MEGA System Oriented IT Architecture

Benutzerhandbuch

MEGA HOPEX V1R2-V1R3
1. Ausgabe (Juli 2015)

Die in diesem Dokument enthaltenen Informationen können jederzeit ohne vorherige Ankündigung geändert werden und stellen in keinem Fall eine Verpflichtung für MEGA International dar.

Diese Veröffentlichung darf weder ganz noch teilweise ohne vorherige schriftliche Erlaubnis von MEGA International für irgendwelche Zwecke reproduziert, gespeichert, übersetzt oder übertragen werden, unabhängig davon, mit welchen Mitteln dies geschieht.

© MEGA International, Paris, 1996 - 2015

Alle Rechte vorbehalten.

MEGAWindows ist eine registrierte Marke von MEGA International.

Windows ist eine registrierte Marke von Microsoft.

Die anderen Marken gehören den jeweiligen Eigentümern.

INDICE

Einleitung 7

Über dieses Handbuch 8
Struktur des Handbuchs 8
Weitere Ressourcen 8
Schreibkonventionen in diesem Handbuch. 9
Stil und Formatierung. 9

Der Arbeitsbereich 11

Verbindung zu MEGA Windows Front-End 12
Verbindung zu MEGA WEB Front-End 14
Allgemeine Präsentation der Benutzeroberfläche 17

Verwendete Konzepte - Einführung 19

Beispiele für die Verwendung. 20
Beispiel eines Anwendungssystems 21
Strukturdiagramm eines Anwendungssystems. 21
Diagramm über Anwendungsstruktur 22
Beispiel einer Architektur logischer Anwendungen 24
Diagramm einer Architektur logischer Anwendungen 24
Das Konzept Data Store 27
Einführung in das Konzept Data Store 27
Verwendungskontexte 28
Beispiel der Funktionsweise von Austauschverträgen und Interaktionen 28
Diagramm des Austauschvertrags (BPMN) 29
Diagramm eines Austauschs 29
Beispiel eines erweiterten Austauschvertrags einer Kommunikation 30

In den Diagrammen beschriebene Elemente 32
Kontextmenü der beschriebenen Elemente 32
Eigenschaften der beschriebenen Elemente 33
Diagramme und ihre Objekte 34
Eigenschaften der verbundenen Objekte 35
Eigenschaften erstellen 35
Vererbung der Eigenschaften 35

Mit MEGA System Oriented IT Architecture modellieren 37

Anwendungssystem beschreiben 38
Anwendungssystem erstellen 38
Komponenten eines Anwendungssystems beschreiben 39
Verwendung eines Anwendungssystems erstellen 40
Anwendungskomponente erstellen 40
Endbenutzer erstellen 41
Beschreibung eines Anwendungssystems ergänzen 41
Zugriff auf die Beschreibung eines Anwendungssystems 41
Anwendungssystem mit einem Systemprozess verbinden 42
Struktur einer Anwendung beschreiben 42
Architektur logischer Anwendungen beschreiben 45
Architektur logischer Anwendungen erstellen 45
Komponenten der Architektur logischer Anwendungen beschreiben. 46
Data Stores verwenden. 48
Physische Data Stores beschreiben 48
Physisches lokales RDB-Data Store erstellen 48
Physisches externes RE-Data Store erstellen 49
Zugriff auf ein physisches Data Store beschreiben 49
Logisches Data Store beschreiben. 50
Logisches lokales Data Store erstellen. 50
Zugriff auf ein logisches Data Store beschreiben 50
Realisierung eines Data Stores 51
Dienst- und Abfragepunkte beschreiben. 52
Konzepte - Einführung. 52
Dienstpunkte 52
Abfragepunkte 53
Dienstpunkt oder Abfragepunkt erstellen 54
Interaktionen verwalten 55
Interaktion erstellen 55
Interaktionspunkt eines Elements definieren 55
Das Prinzip 55
Merkmale des Interaktionspunktes eines Elements definieren 56

Mit Service Design modellieren 57

Austauschvertrag beschreiben 58
Austauschvertrag ab einer Interaktion erstellen 59
Diagramm eines Austauschvertrags erstellen (BPMN) 60
Verwendeten Austausch oder verwendeten Austauschvertrag definieren 60
Ereignisse erstellen 61
Sequenzflüsse bearbeiten 61
Die Gateways 62
Gateways am Ausgang einer Etappe 62
Die Gateways am Eingang einer Etappe 63
Gateway erstellen 63
Gateway ändern 63
Austausch beschreiben 65
Austausch erstellen 66
Austausch beschreiben 66
Austauschdiagramm (BPMN) erstellen 66
Nachrichtenfluss mit Inhalt erstellen 66

Standardberichte von MEGA System Oriented IT Architecture verwenden . . . 69

Übereinstimmung der Austausche zwischen Agenten 70
Ausgleich der Austausche zwischen Agenten 72
Kreuztabelle der Interaktionen von Diensten 74
Der Bericht 74
Details des Austauschs 75
Berichtparameter 76

Glossar 77

Sommaire

EINLEITUNG

MEGA System Oriented IT Architecture ist ein Werkzeug von **MEGA International**, mit dem U8nternehmen und andere Einrichtungen IT-Architekturen mit einer serviceorientierten Methode darstellen können.

Dieses Handbuch soll erklären, wie diese Funktionalitäten genutzt werden, um Änderungen an Ihrem Informationssystem vornehmen zu können.

ÜBER DIESES HANDBUCH

Dieses Handbuch erklärt, wie Sie mit **MEGA System Oriented IT Architecture** die IT-Architektur Ihres Unternehmens anhand einer serviceorientierten Methode beschreiben können.

> ☛ *Die Unterschiede zwischen **MEGA Windows Front-End** und **MEGA Web Front-End** werden ggf. für jede einzelne Funktionalität beschrieben.*

Struktur des Handbuchs

Das Handbuch **MEGA System Oriented IT Architecture** setzt sich aus folgenden Kapiteln zusammen:

- "Arbeitsbereich entdecken", Seite 15, beschreibt die Schritte vor der Nutzung dieser Lösung und stellt die Benutzeroberfläche abhängig vom verbundenen Benutzer vor.
- "Verwendete Konzepte - Einführung", Seite 19, gibt einen Überblick über die grundlegenden Konzepte von **MEGA System Oriented IT Architecture**.
- "Mit MEGA System Oriented IT Architecture modellieren", Seite 37, erklärt, wie die Architektur Ihres Unternehmens dargestellt werden kann.
- "Mit Service Design modellieren", Seite 57, erklärt, wie die Austauschverträge zwischen den Komponenten Ihrer IT-Architektur angegeben werden.
- "Standardberichte von MEGA System Oriented IT Architecture verwenden", Seite 69erläutert, wie die Beziehungen zwischen den Komponenten eines Unternehmens mit **MEGA System Oriented IT Architecture** analysiert werden können.
- "Glossar", Seite 77, fasst die Definitionen der wichtigsten Konzepte des Handbuchs **MEGA System Oriented IT Architecture** zusammen.

Weitere Ressourcen

Dieses Handbuch wird ergänzt durch:

- das Handbuch **MEGA Common Features**, das die grundlegenden gemeinsamen Funktionalitäten aller Produkte und Lösungen von **MEGA** beschreibt.

  > ☛ *Es kann nützlich sein, dieses Handbuch aufzurufen, um eine allgemeine Darstellung der Benutzeroberfläche zu erhalten.*

- das Administrationshandbuch MEGA Supervisor**.**
- die erweiterten technischen Funktionen, die in dem Handbuch **HOPEX Studio** behandelt werden.

Schreibkonventionen in diesem Handbuch

Stil und Formatierung

☛ *Anmerkung zu den vorherigen Punkten.*

📖 *Definition der verwendeten Begriffe.*

☺ *Tipp, der dem Benutzer das Leben erleichtern kann.*

Kompatibilität mit vorherigen Versionen.

Zu vermeiden.

Wichtiger Hinweis zur Vermeidung von Bedienungsfehlern.

Befehle werden wie folgt dargestellt: **Datei > Öffnen**.

Namen von Produkten und technischen Modulen werden wie folgt dargestellt: **MEGA**.

DER ARBEITSBEREICH

Dieses Kapitel macht Sie mit der Verwendung von **MEGA System Oriented IT Architecture** vertraut und stellt die einzelnen Schritte vor, die vor der Nutzung der Lösung durchzuführen sind.

An dieser Stelle werden folgende Punkte erklärt:

- ✓ "Verbindung zu MEGA Windows Front-End", Seite 12
- ✓ "Verbindung zu MEGA WEB Front-End", Seite 14
- ✓ "Allgemeine Präsentation der Benutzeroberfläche", Seite 17

VERBINDUNG ZU MEGA WINDOWS FRONT-END

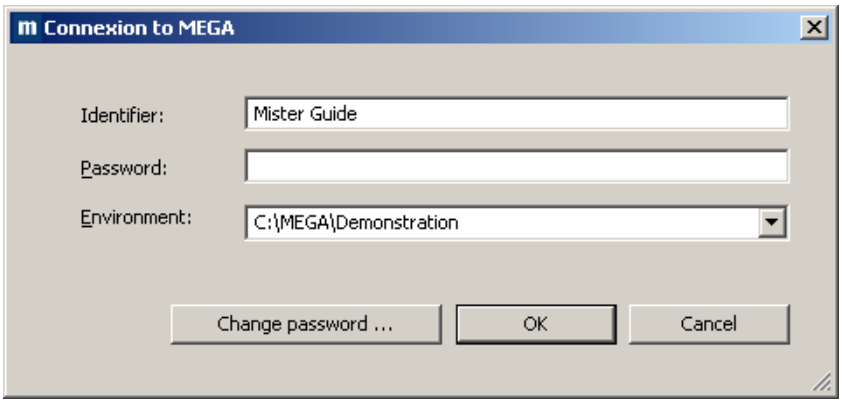
MEGA Windows Front-End Front-End ist eine Anwendung von MEGA, auf die im Windows-Desktop zugegriffen werden kann.

Damit Sie sich mit **MEGA** vertraut machen können, werden in diesem Handbuch Beispiele aus dem Repository "MEGA (Tutorial)" der Umgebung "Demonstration" verwendet. Der Benutzer „Mister Guide" wurde erstellt, um Sie als Anfänger durch **MEGA** zu führen.

Um **MEGA** zu starten, gehen Sie wie folgt vor:

1. Doppelklicken Sie auf das Symbol **MEGA**

.
Das Identifikationsfenster wird angezeigt.



2. Wählen Sie im Feld **Umgebung** Ihre Arbeitsumgebung aus.

   ☛ *Sollten Sie keinen Zugriff auf eine Umgebung haben, wird die Umgebung automatisch berücksichtigt. Das Feld zur Auswahl der Umgebung ist abgeblendet.*

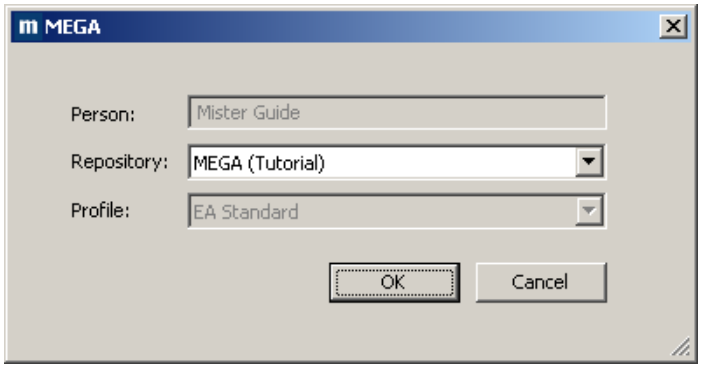
   ☛ *Weitere Informationen zu den Umgebungen und den Benutzern finden Sie im Kapitel **Umgebungen verwalten** und **Benutzer verwalten** im Handbuch **MEGA Administration**.*

3. Geben Sie im Feld **ID** Ihren Benutzernamen ein.

   ☛ *Wenn Sie die Umgebung "Demonstration" gewählt haben, können Sie mit der ID "Mister Guide" arbeiten.*

4. Geben Sie (ggf.) im Feld **Kennwort** Ihr Kennwort ein (sofern notwendig).

5. - Haga clic en OK.
   Wenn Sie authentifiziert sind, wird das Fenster zur Auswahl eines Arbeitsbereichs angezeigt.



   Das Feld **Person** wird automatisch definiert und gibt den Namen der Person an, die mit der ID verbunden ist, die im Fenster zur Verbindung angegeben wurde.

6. Wählen Sie im Feld **Repository** das Arbeits-Repository aus.

   ☛ *Im Dropdown-Menü wird die Liste der in der Umgebung verfügbaren Repositorys angezeigt. Wenn Sie in Ihrem Unternehmen beginnen Daten zu modellieren, ist es besser, ein neues Repository in einer neuen Umgebung zu erstellen.*

   ☛ *Sollten Sie nur auf ein Repository Zugriff haben, wird das Repository automatisch berücksichtigt. Das Feld zur Auswahl des Repositorys ist abgeblendet.*

7. Klicken Sie im Feld **Profil** oder **Geschäftsrolle** auf den Pfeil und wählen Sie das Profil oder die Rolle aus, mit der bzw. dem Sie arbeiten möchten.

   ☛ *Sollten Sie nur ein Profil oder eine Rolle haben, wird das Profil oder die Rolle automatisch berücksichtigt. Das Feld zur Auswahl des Profils oder der Rolle wird nicht angezeigt.*

8. Klicken Sie auf **OK**.
   Ihr Desktop wird angezeigt.

VERBINDUNG ZU MEGA WEB FRONT-END

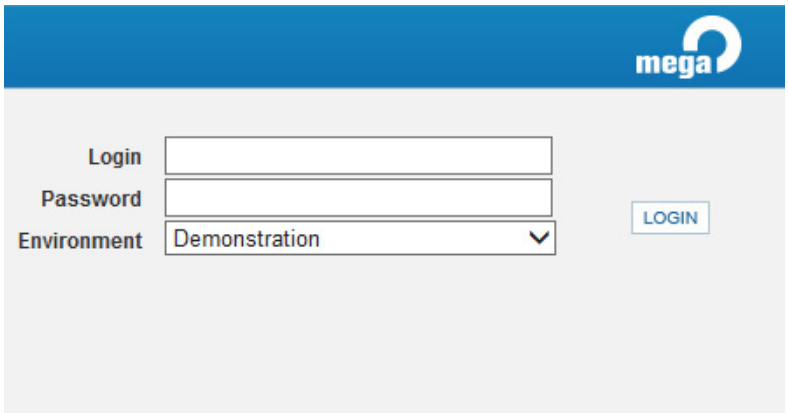
MEGA Web Front-End Front-End ist eine Anwendung von MEGA, auf die in einem Internet-Browser zugegriffen werden kann.

Um sich mit einer **MEGA** -Anwendung im Web zu verbinden, gehen Sie wie folgt vor:

1. Starten Sie die Anwendung **MEGA** unter ihrer HTTP-Adresse.

   ☛ *Wenn Sie die Adresse nicht kennen, wenden Sie sich an Ihren Administrator.*

   Daraufhin wird die Verbindungsseite geöffnet.



2. Geben Sie auf der Verbindungsseite im Feld **Login** Ihre ID ein.
3. (Wenn Sie ein Kennwort besitzen) Geben Sie im Feld **Password** Ihr Kennwort ein.

   ☛ *Wenn Sie Ihr Kennwort vergessen haben, klicken Sie auf* Lost password *(unter dem Verbindungsfenster).*

4. Klicken Sie im Feld **Environment** auf den Pfeil und wählen Sie die Arbeitsumgebung aus.

   ☛ *Sollten Sie keinen Zugriff auf eine Umgebung haben, wird die Umgebung automatisch berücksichtigt. Das Feld zur Auswahl der Umgebung wird nicht angezeigt.*

5. Klicken Sie auf **LOGIN**.
   Wenn Sie authentifiziert wurden, wird ein neues Fenster angezeigt.
6. Klicken Sie im Feld **Repository** auf den Pfeil und wählen Sie das Arbeits-Repository aus.

   ☛ *Wenn Sie nur Zugriff auf ein Repository haben, wird dieses Repository automatisch berücksichtigt.*

7. Klicken Sie im Feld **Business Role** oder **Profile** auf den Pfeil und wählen Sie die Geschäftsrolle oder das Profil aus, mit der bzw. dem Sie arbeiten möchten.

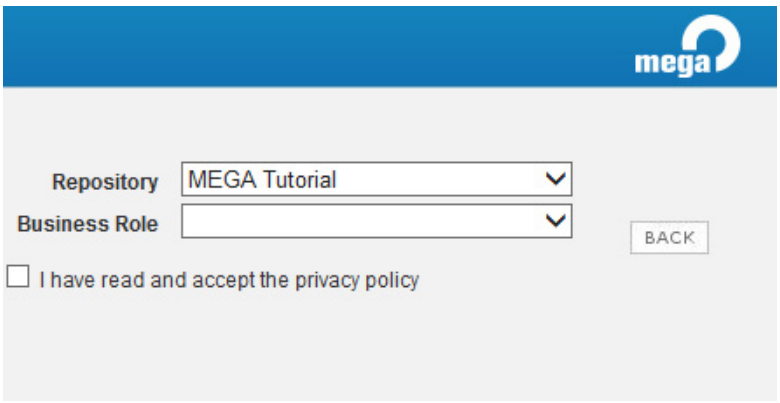
   ☛ *Wenn Sie nur eine Rolle oder ein Profil haben, wird diese Rolle bzw. dieses Profil automatisch berücksichtigt.*

   ☛ *In **MEGA Administration** wird das Feld **Profile** statt **Business Role** angezeigt, wenn in den Optionen der Umgebung (Optionen/ Installation/Benutzerverwaltung) die Option "Verwaltung der Zuweisung von Geschäftsrollen an Personen" nicht ausgewählt wurde.*

   ☛ *Das ausgewählte Profil bzw. die Geschäftsrolle definiert die Anwendungen, auf die Sie zugreifen können.*

8. Klicken Sie im Feld **Application** auf den Pfeil und wählen Sie die Anwendung aus, mit der Sie sich verbinden möchten.

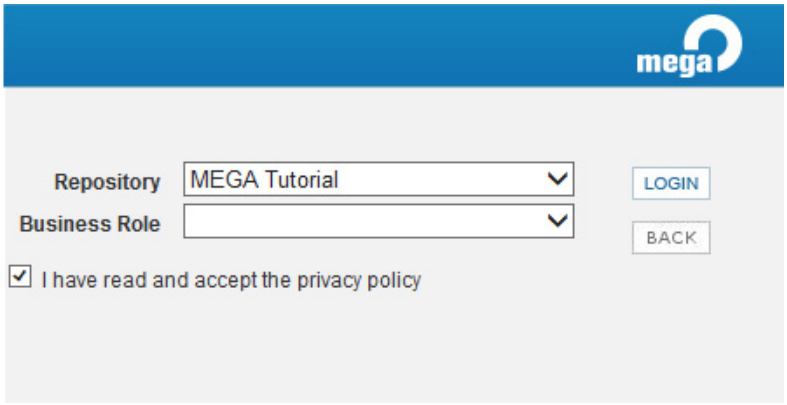
   ☛ *Sollten Sie Zugriff nur auf eine Anwendung mit dem ausgewählten Profil bzw. der Geschäftsrolle haben, wird das Profil/die Rolle automatisch berücksichtigt. Das Feld zur Auswahl der Anwendung wird nicht angezeigt.*



9. Klicken Sie auf Privacy Policy (unter dem Fenster zur Anmeldung) und lesen Sie die Vorschriften hinsichtlich der Datenschutzpolitik durch. Wählen Sie anschließend **I have read and accept the privacy policy** aus.
   Die Schaltfläche **LOGIN** wird angezeigt.

   ☛ *Wenn Sie die Vorschriften hinsichtlich der Datenschutzpolitik gelesen und akzeptiert haben, wird automatisch ein Zertifikat mit Ihrer*

Person verbunden. Zu diesem Schritt werden Sie später nicht mehr aufgefordert.



10. Klicken Sie auf **LOGIN**.

☛ *Klicken Sie auf **BACK**, wenn Sie zum Authentifizierungsfenster zurückkehren möchten.*

Die Home-Page der Web-Anwendung wird angezeigt und eine Sitzung wird geöffnet.

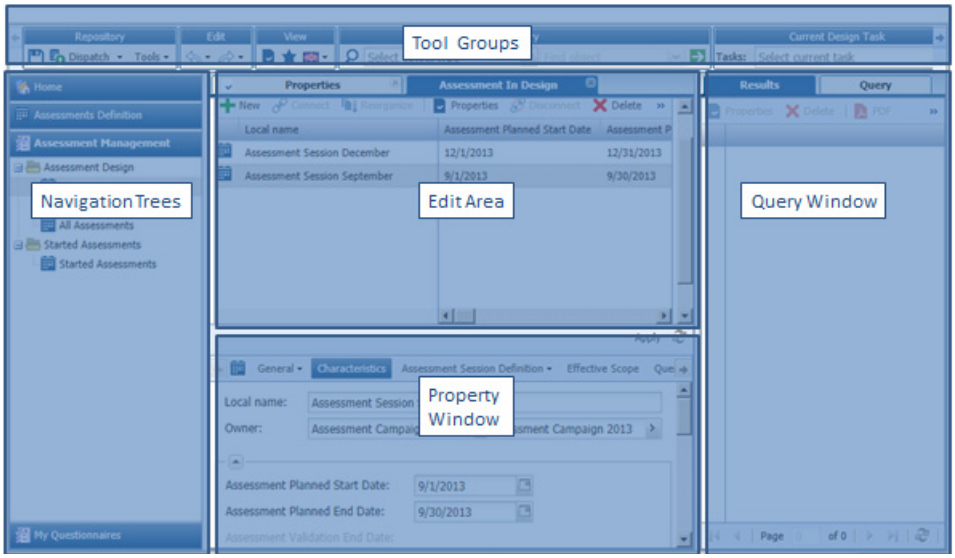
☛ *Weitere Informationen zur Definition der Benutzer und Rollen finden Sie im Handbuch **MEGA Administration - Supervisor**, Kapitel "Benutzer verwalten".*

ALLGEMEINE PRÄSENTATION DER BENUTZEROBERFLÄCHE

MEGA System Oriented IT Architecture setzt sich aus folgenden Zonen zusammen:

- Im oberen Teil, die ***Werkzeuggruppen*** (***Repository***, ***Bearbeiten***, ***Ansicht***, und ***Abfrage***, ***Divers***), mit denen Sie die Befehle schnell finden, die Sie zur Ausführung einer Aufgabe brauchen.
- Links befinden sich ***Navigationsbaumstrukturen***, mit denen Sie auf Objekte im Repository zugreifen können.
- In der Mitte befindet sich eine ***Bearbeitungszone***, in der das Eigenschaftsfenster des aktuellen Objekts angezeigt werden kann.

Sie können auch das ***Abfragefenster*** anzeigen und verankern.



☛ *Weitere Informationen zu den Standardfunktionalitäten von **MEGA** finden Sie im Handbuch **MEGA Common Features**.*

VERWENDETE KONZEPTE - EINFÜHRUNG

MEGA System Oriented IT Architecture ist ein Werkzeug zur Beschreibung der serviceorientierten Architektur, mit denen die Kommunikation zwischen den einzelnen Architekturen leichter gestaltet werden soll. Bei der Beschreibung der Architekturen werden besondere Konzepte verwendet, mit denen sich das Produkt allgemeiner verwenden lässt.

- ✓ "Beispiele für die Verwendung", Seite 20
- ✓ "In den Diagrammen beschriebene Elemente", Seite 32
- ✓ "Diagramme und ihre Objekte", Seite 34
- ✓ "Eigenschaften der verbundenen Objekte", Seite 35

BEISPIELE FÜR DIE VERWENDUNG

Mit den von **MEGA System Oriented IT Architecture** aufgeführten Funktionalitäten für die Modellierung serviceorientierter Architekturen können dargestellt werden:

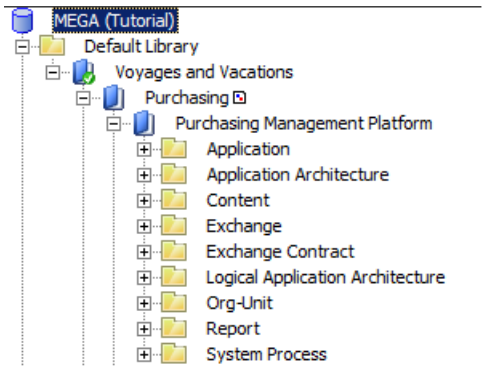
- Hardware- und organisatorische Komponenten, die für die korrekte Funktionsweise des IT-Systems notwendig sind
- Interaktionen zwischen den Komponenten
- die Dienste, die im Rahmen der modellierten Architektur angeboten und genutzt werden

Die beschriebenen Komponenten können einer Modellierungsmethodologie folgen:

- aufsteigend, von einer detaillierten zu einer konzeptuellen Darstellung
- absteigend, von einer konzeptuellen zu einer detaillierten Darstellung

Diese Modellierungsfunktionen beruhen auf dem Beispiel eines Anwendungssystems für die Bearbeitung von Einkäufen, die mit einer absteigenden Methode ab folgenden Diagrammen vorgestellt wird:

- Diagramm einer Anwendungssystemstruktur
- Diagramm einer Anwendungsstruktur
- Diagramm der Architektur einer logischen Anwendung logischen Anwendung



Beispiel eines Anwendungssystems

Strukturdiagramm eines Anwendungssystems

Die Komponenten eines *Anwendungssystems* werden in einem Strukturdiagramm eines Anwendungssystems beschrieben, das die interne Struktur des Anwendungssystems beschreibt:

- die angebotenen oder erforderlichen Dienste
- die übernommenen Prozesse
- die Komponenten und ihre Interaktionen
- die Endbenutzer in Interaktion mit den Anwendungskomponenten
- Data Stores verwenden

> *Ein Anwendungssystem ist eine Montage aus anderen Anwendungssystemen, Anwendungen und Endbenutzern, die mit den Anwendungskomponenten zusammenwirken, um eine oder mehrere Funktionalitäten durchzuführen.*

Das Diagramm beschreibt das Anwendungssystem, das der Bearbeitungsprozess der Einkaufsanträge entspricht.

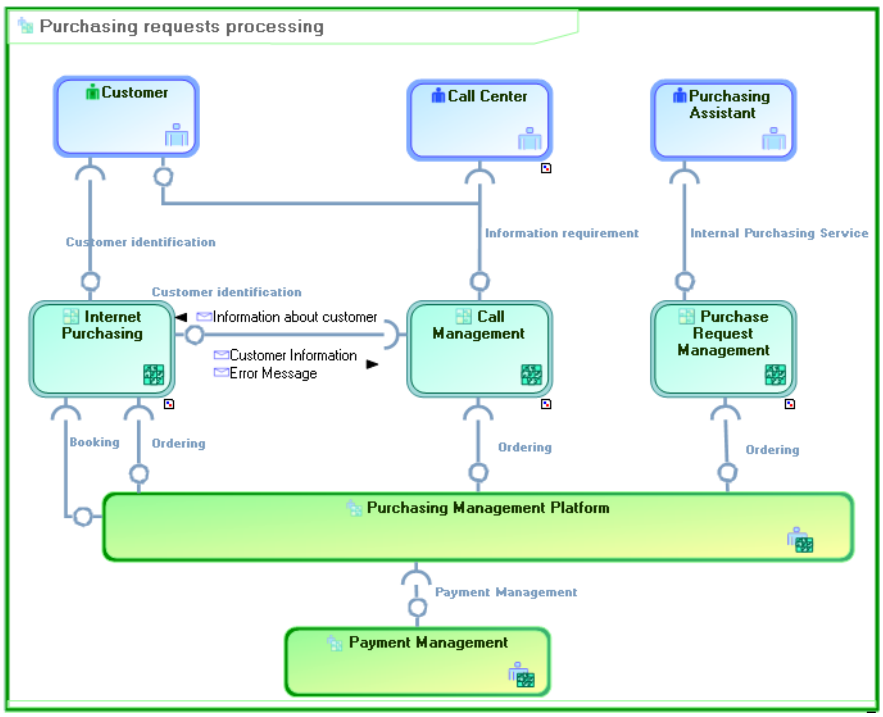


Diagramm der Struktur des Anwendungssystems "Bearbeitung der Einkaufsanträge"

Die Einkaufsanträge können vom Kunden direkt über eine eCommerce-Anwendung formuliert werden, oder aber indirekt,

über ein Call Center. Die internen Einkaufsanträge erfolgen durch einen Einkaufsassistenten.

Bei der Architektur der Bearbeitung von Einkaufsanträgen kommen die Dienste der Anwendungssysteme der "Plattform der Einkaufsverwaltung" und "Zahlungsmanagement" zum Einsatz.

Das Diagramm besteht aus:

- zwei Elementen vom Typ *Verwendung des Anwendungssystems*, die die beiden verwendeten Anwendungssysteme darstellen.

  📖 *Die Nutzung eines Anwendungssystems beschreibt die Rolle eines Anwendungssystems in einer zusammengesetzten Beziehung mit einem übergeordneten Anwendungssystem.*

- drei *Anwendungskomponenten*, die die im beschriebenen Kontext des Anwendungssystems verwendeten Anwendungen darstellen.

  📖 *Eine IT-Service-Komponente stellt die Tatsache dar, dass der verwendete IT-Service eine Rolle in der beschriebenen Anwendung spielt.*

- drei Komponenten des Typs *Endbenutzer*, die die Kategorien der Mitwirkenden im beschriebenen Anwendungssystem darstellen.

  📖 *Ein Endbenutzer stellt eine OrgEinheit dar, die an den Grenzen eines Anwendungssystems oder einer Architektur logischer Anwendungen mitwirkt.*

- *Interaktionen* zwischen den Komponenten, die die Dienstanträge darstellen.

  📖 *Eine Interaktion ist ein Vertrag, der in einem bestimmten Kontext zwischen zwei autonomen Entitäten innerhalb oder außerhalb eines Unternehmens abgeschlossen wird. Diese Entitäten können Organisationseinheiten, Aktivitäten oder Prozesse des Unternehmens sein, sowie externe Orgeinheiten eines Unternehmens. Der Inhalt dieses Vertrages wird von einem Austauschvertrag beschrieben.*

Diagramm über Anwendungsstruktur

In einer Anwendung können Sie auf das Diagramm einer Anwendungsstruktur zugreifen, das die wichtigsten Elemente für die Funktionsweise der Anwendung beschreibt.

📖 *In einem Diagramm der Anwendungsstruktur werden die Komponenten auf der ersten Ebene einer Anwendung, die*

Zugriffspunkte (Dienst- und Abfragepunkt) und die Verbindungen zwischen den Komponenten grafisch dargestellt.

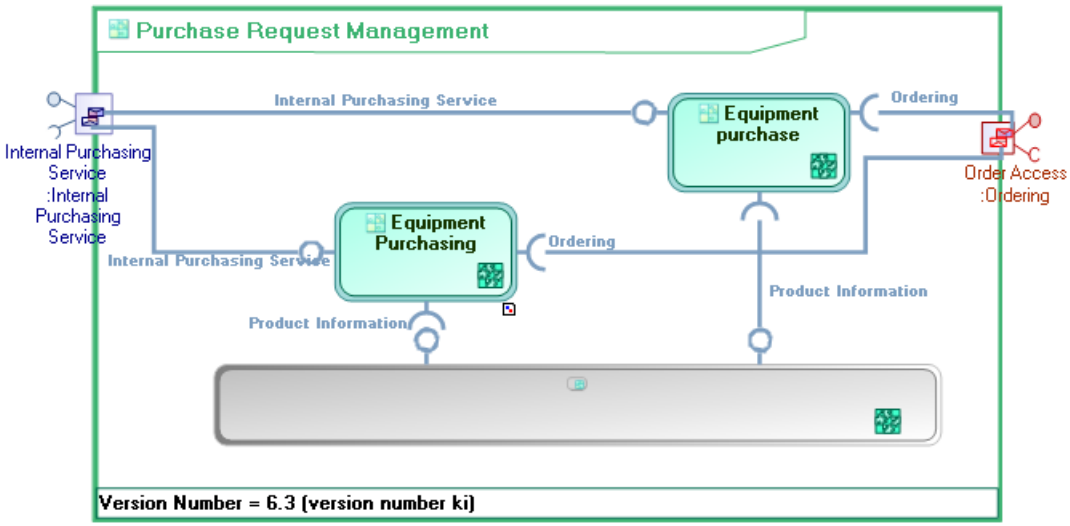


Diagramm der Anwendungsstruktur "Verwaltung der Einkaufsanträge"

```
Die Anwendung zur Verwaltung der Einkaufsanträge wird nicht
für interne Einkäufe verwendet und stützt sich auf zwei
spezielle Anwendungen: eine für den Einkauf von
Gebrauchsartikeln und eine andere für die
Materialbeschaffung. Die beiden Anwendungen verwenden
Microsoft Excel.
```

☛ *Die Versionsnummer im unteren Bereich des Rahmens ist eine Eigenschaft der Anwendung. Weitere Informationen finden Sie unter "Eigenschaften der beschriebenen Elemente", Seite 33.*

Dieses Diagramm besteht aus folgenden Elementen:

- Zwei *Anwendungskomponenten*, die die verwendeten Anwendungen darstellen

  🕮 *Eine IT-Service-Komponente stellt die Tatsache dar, dass der verwendete IT-Service eine Rolle in der beschriebenen Anwendung spielt.*

- einer *Komponente einer Plattformanwendung*, die im Beispiel mit der Excel-Anwendung verknüpft ist.

  🕮 *Komponente einer Plattformanwendung stellt die Tatsache dar, dass die verwendete Technologie in der beschriebenen Anwendung die Rolle einer Plattform übernimmt. Beispielsweise ist Tomcat eine Plattformkomponente für SageCRM.*

- *Interaktionen* zwischen den Komponenten, die die Dienstanträge darstellen.
  Wichtig: Der Dienstpunkt "Zugang interner Einkauf" ist aktiviert. "Zugang Bestellung" aktiviert den Dienst der Bestellungen, wenn die Einkaufsanträge freigegeben wurden.

  🕮 *Eine Interaktion ist ein Vertrag, der in einem bestimmten Kontext zwischen zwei autonomen Entitäten innerhalb oder außerhalb eines Unternehmens abgeschlossen wird. Diese Entitäten können*

Organisationseinheiten, Aktivitäten oder Prozesse des Unternehmens sein, sowie externe Orgeinheiten eines Unternehmens. Der Inhalt dieses Vertrages wird von einem Austauschvertrag beschrieben.

Beispiel einer Architektur logischer Anwendungen

📖 *Die Architektur logischer Anwendungen ist eine Montage aus anderen Architekturen, logischen Anwendungen und Endbenutzern, die mit den Anwendungskomponenten zusammenwirken, um eine oder mehrere Funktionalitäten durchzuführen.*

📖 *Eine logische Anwendung ist das Teilelement einer Anwendung, das dem Endbenutzer dieser Anwendung im Rahmen seiner Arbeit zur Verfügung gestellt wird. Die logischen Anwendungen stellen die Aufteilung der Anwendungen in Form funktionaler elementarer Einheiten dar. Eine logische Anwendung ist eine kohärente und nicht teilbare Bearbeitungseinheit, die eine Gruppe von Nachrichten und Ereignissen koordiniert, um im Informationssystem eine Task durchzuführen. Im Anwendungssystem entspricht die logische Anwendung einer funktionellen Einheit, die weitaus elementarer ist.*

Diagramm einer Architektur logischer Anwendungen

Die Komponenten der *Architektur logischer Anwendungen* werden in einem Diagramm einer Architektur logischer Anwendungen beschrieben, das Folgendes darstellt:

- die angebotenen oder erforderlichen Dienste
- die übernommenen Prozesse, die Komponenten und ihre Interaktionen
- die Endbenutzer in Interaktion mit den Anwendungskomponenten

Folgendes Diagramm beschreibt die Architektur einer logischen Anwendung für den Einkauf per eCommerce, die den Kunden angeboten wird.

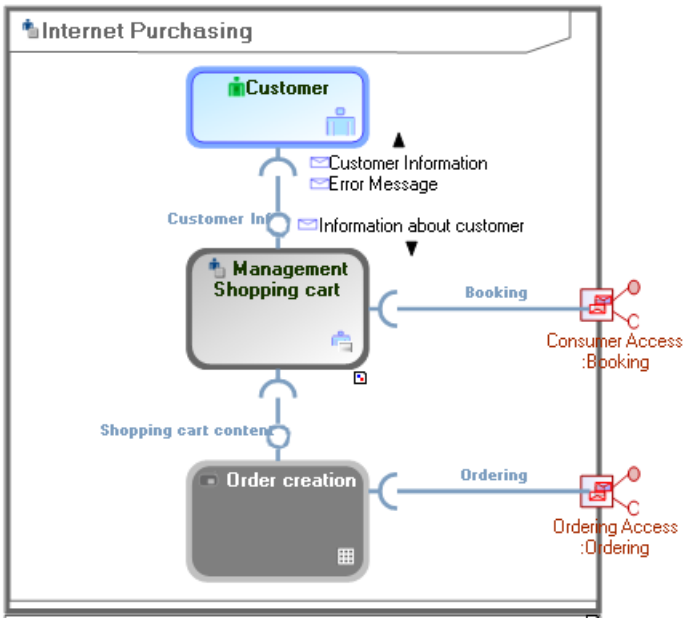


Diagramm einer Architektur logischer Anwendungen "Einkauf per eCommerce"

Die von den Kunden formulierten Anträge werden von einer Architektur logischer Anwendungen "Warenkorb des Kunden verwalten" bearbeitet. Die logische Anwendung "Bestellung

erstellen" wird anschließend im Kontext dieser Architektur verwendet.

Das Diagramm einer Architektur logischer Anwendungen, das den Warenkorb des Kunden verwaltet, führt zwei Abfragepunkte für "Reservieren" oder "Bestellen" auf.

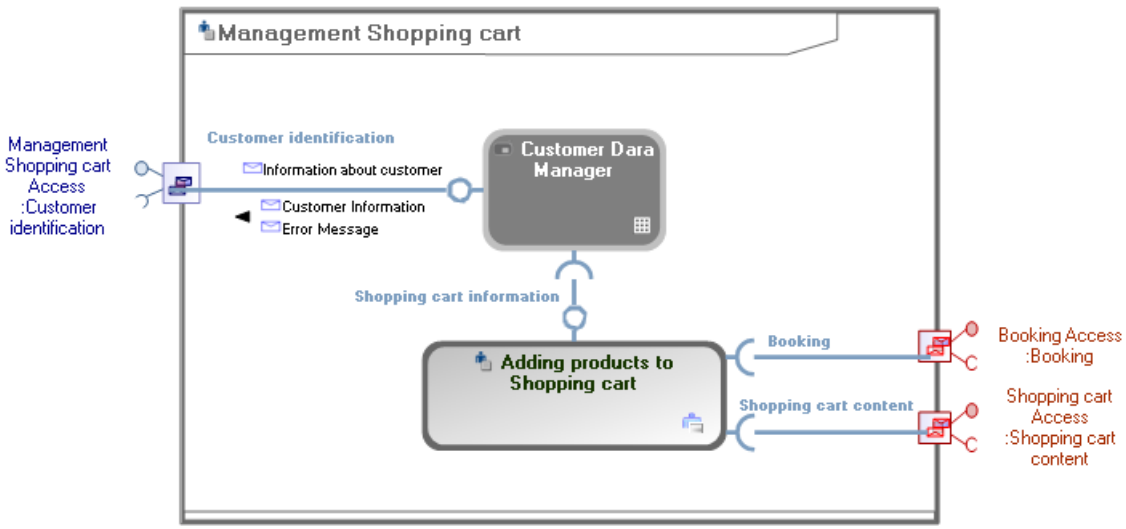


Diagramm einer Architektur logischer Anwendungen "Warenkorb des Kunden verwalten"

Diese Diagramme bestehen aus:

- einer Komponente des Typs *Endbenutzer*, die in diesem Beispiel die Kunden darstellt.

  > 🕮 *Ein Endbenutzer stellt eine OrgEinheit dar, die an den Grenzen eines Anwendungssystems oder einer Architektur logischer Anwendungen mitwirkt.*

- mehreren *Verwendungen logischer Anwendungen*, die die beiden verwendeten Architekturen logischer Anwendungen darstellen.

  > 🕮 *Die Nutzung einer Architektur logischer Anwendungen beschreibt die Rolle einer Architektur logischer Anwendungen in einer zusammengesetzten Beziehung mit einer übergeordneten Architektur logischer Anwendungen.*

- Die *Komponenten einer logischen Anwendung*, die die verwendeten logischen Anwendungen im Kontext der beschriebenen Architektur einer logischen Anwendung darstellen.

  > 🕮 *Die Komponente einer logischen Anwendung beschreibt die Rolle einer logischen Anwendung in einer zusammengesetzten Beziehung mit einer übergeordneten logischen Anwendung oder einer übergeordneten Architektur logischer Anwendungen.*

- *Interaktionen* zwischen den Komponenten, die die Dienstanträge darstellen.

  > 🕮 *Eine Interaktion ist ein Vertrag, der in einem bestimmten Kontext zwischen zwei autonomen Entitäten innerhalb oder außerhalb eines Unternehmens abgeschlossen wird. Diese Entitäten können Organisationseinheiten, Aktivitäten oder Prozesse des Unternehmens*

sein, sowie externe Orgeinheiten eines Unternehmens. Der Inhalt dieses Vertrages wird von einem Austauschvertrag beschrieben.

Das Konzept Data Store

Ein Data Store bietet einen Mechanismus an, mit dem Daten aktualisiert oder abgerufen werden, die auch über die Reichweite des aktuellen Prozesses hinaus bestehen bleiben. In einem Data Store werden die Daten der Flüsse am Eingang gespeichert und über einen oder mehrere Flüsse am Ausgang weitergeleitet.

Einführung in das Konzept Data Store

Je nach beschriebenem Kontext schlägt **MEGA System Oriented IT Architecture** die Verwendung von RE- oder RDB-Data Stores vor.

Ein Relationale Datenbank-Data Store ist ein Data Store in Form von Tabellen oder Tabellenansichten.

Ein Entität/Relation-Data Store beschreibt die Daten einer Datenbank in Form von Entitäten oder Klassen.

Ein Data Store referenziert eine **Struktur des Datenbestands**.

Die Struktur eines RDB-Datenbestands wird von einem Paket gehalten. Die Struktur eines RE-Datenbestands wird von einer Datenbank gehalten.

Wenn Sie ein Anwendungssystem beschreiben, können nur die physischen Data Stores verwendet werden.

Ein physisches Data Store stellt die Implementierung eines logischen Data Stores dar.

Weitere Informationen finden Sie unter "Physische Data Stores beschreiben", Seite 48.

Wenn Sie die Architektur einer logischen Anwendung beschreiben, können nur die logischen Data Stores verwendet werden.

Ein logisches Data Store stellt die Verwendung der Daten über Anwendungssysteme dar, ohne zu berücksichtigen, wie der Zugriff konkret implementiert wird.

Weitere Informationen finden Sie unter "Logisches Data Store beschreiben", Seite 50.

In **MEGA System Oriented IT Architecture** kann zwischen den lokalen Data Stores eines Systems (oder eine Architektur) der externen Data Stores unterschieden werden, die an der Grenze der Diagramme positioniert werden.

Ein lokales Data Store ist ein Data Store, das ausschließlich innerhalb des beschriebenen Systems verwendet wird.

Ein externes Data Store ist ein Data Store, das innerhalb und außerhalb des beschriebenen Systems verwendet wird.

Verwendungskontexte

Data Stores können in folgenden Diagrammen beschrieben werden:

- Strukturdiagramme eines Anwendungssystems
- Diagramme über Anwendungsstruktur
- Diagramme der Architektur einer logischen Anwendung

☛ *Weitere Informationen finden Sie unter "Data Stores verwenden", Seite 48.*

Beispiel der Funktionsweise von Austauschverträgen und Interaktionen

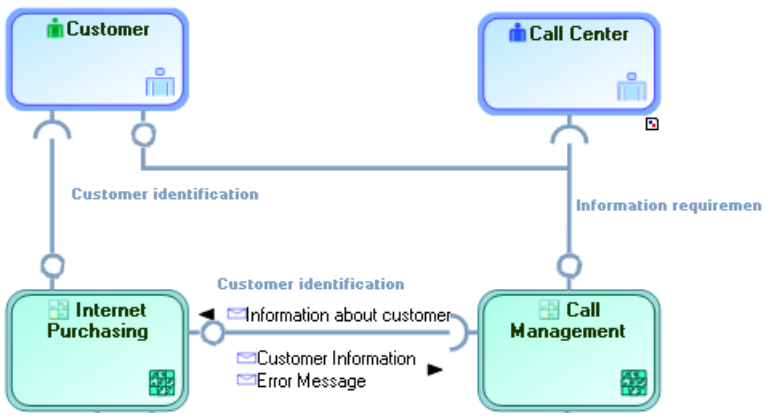
Eine *Interaktion* stellt den Informationsaustausch zwischen den Komponenten einer Architektur dar.

> 🕮 *Eine Interaktion ist ein Vertrag, der in einem bestimmten Kontext zwischen zwei autonomen Entitäten innerhalb oder außerhalb eines Unternehmens abgeschlossen wird. Diese Entitäten können Organisationseinheiten, Aktivitäten oder Prozesse des Unternehmens sein, sowie externe Orgeinheiten eines Unternehmens. Der Inhalt dieses Vertrages wird von einem Austauschvertrag beschrieben.*

Der Inhalt einer Interaktion wird von einem *Austauschvertrag* beschrieben.

> 🕮 *Ein Austauschvertrag ist ein Vertragsmodell zwischen zwei Organisationseinheiten. Dieser Vertrag wird von den Austauschen zwischen der Rolle eines Initiators und einer oder mehreren Rollen der Beitragsleistenden beschrieben.*

Im Diagramm des Anwendungssystems "Bearbeitung der Einkaufsanträge" werden zwei Interaktionen verwendet.



Interaktionen im Diagramm der Struktur eines Anwendungssystems "Bearbeitung der Einkaufsanträge"

```
Die Kunden können in einer eCommerce-Anwendung Bestellungen
aufgeben, oder aber über ein Call Center, das selbst eine
```

eCommerce-Anwendung verwendet, um den Dienst der Kundenidentifikation zu erhalten.

Ein Kunde, der seine Bestellung über ein Call Center aufgibt, kann andere Anträge als Bestellungen senden.

Diagramm des Austauschvertrags (BPMN)

Das Diagramm eines Austauschvertrags, das mit dem Austauschvertrag "Identifikation des Kunden" verknüpft ist, beschreibt in einer BPMN-Notation die durchgeführten Vorgänge.

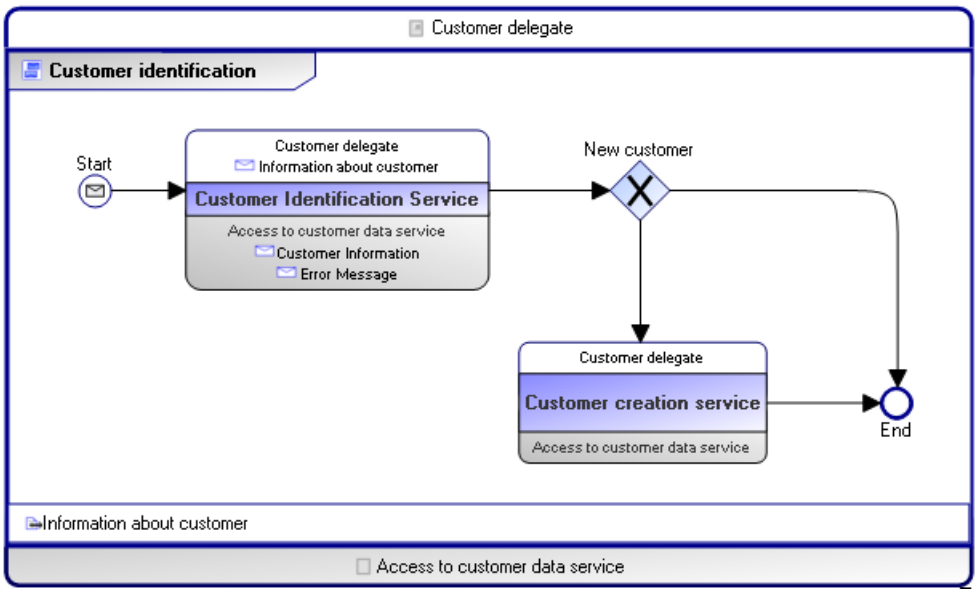


Diagramm eines Austauschvertrags (BPMN) "Identifikation des Kunden"

Der Ablauf der Identifikation des Kunden beginnt mit der Suche des Kunden. Wenn der Kunde gefunden ist, gibt der Austauschvertrag die Auskünfte über den Kunden zurück. Andernfalls wird ein Austauschvertrag zur Erstellung eines Kunden aktiviert.

Die Etappen des Ablaufs werden mit *verwendeten Austauschen* dargestellt.

Diagramm eines Austauschs

Mit der Option **MEGA Service Design** wird ein *verwendeter Austausch* mit einem *Austausch* verknüpft.

> 🕮 *Ein verwendeter Austausch entspricht der Verwendung eines Austausches in einem Austauschvertrag.*
>
> 🕮 *Ein Austausch beschreibt den Austausch von Nachrichtenflüssen zwischen den Teilnehmern.*

Ein *Austausch* wird von einem Diagramm eines Austauschs beschrieben, das die Folgen ausgetauschter Nachrichten darstellt.

> *Ein Austausch beschreibt den Austausch von Nachrichtenflüssen zwischen den Teilnehmern.*

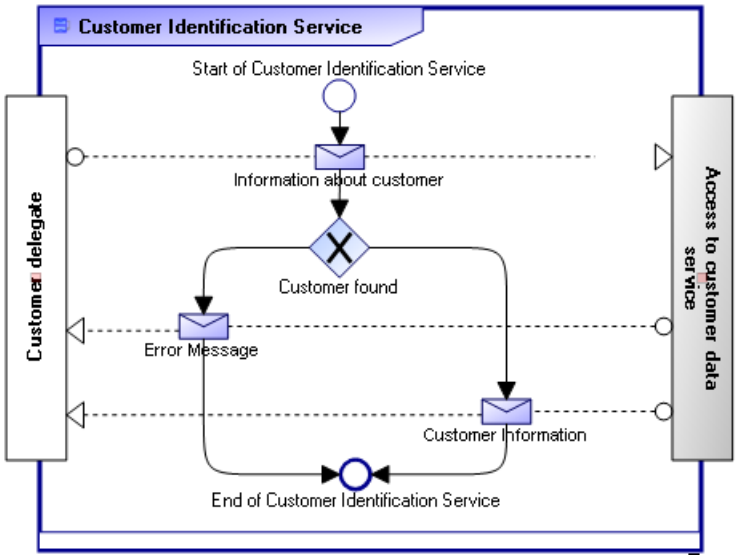


Diagramm eines Austauschs "Dienst Kundenidentifikation"

```
Der Ablauf des Dienstes der Kundenidentifikation beginnt
mit dem Versand von Informationen, mit denen der Kunde
identifiziert wird. Wenn der Kunde nicht gefunden wird,
wird eine Fehlermeldung ausgegeben. Andernfalls werden die
Auskünfte über den Kunden gesendet (Kundenidentifikation,
Status der Bestellungen usw.).
```

Beispiel eines erweiterten Austauschvertrags einer Kommunikation

Mit der Option **MEGA Service Design** wird ein Austauschvertrag mit einer Folge einzelner dargestellter Etappen beschrieben, die wie folgt dargestellt werden:

- entweder mit *verwendeten Austauschverträgen*
- oder mit *verwendeten Austauschen*

> *Ein Austauschvertrag ist ein Vertragsmodell zwischen zwei Organisationseinheiten. Dieser Vertrag wird von den Austauschen zwischen der Rolle eines Initiators und einer oder mehreren Rollen der Beitragsleistenden beschrieben.*

> *Ein verwendeter Austausch entspricht der Verwendung eines Austausches in einem Austauschvertrag.*

Die Rollen des Austauschvertrags, die am Rahmen aufgeführt werden, stellen die mitwirkenden Personen dar:

- Kunde/Lieferant oder
- Sender/Empfänger

Ein Austauschvertrag kann auch von mehr als zwei mitwirkenden Personen beschrieben werden. In diesem Fall handelt es sich bei einer Rolle um den Initiator des Austauschvertrags. Die anderen Mitwirkenden sind die Beitragsleistenden.

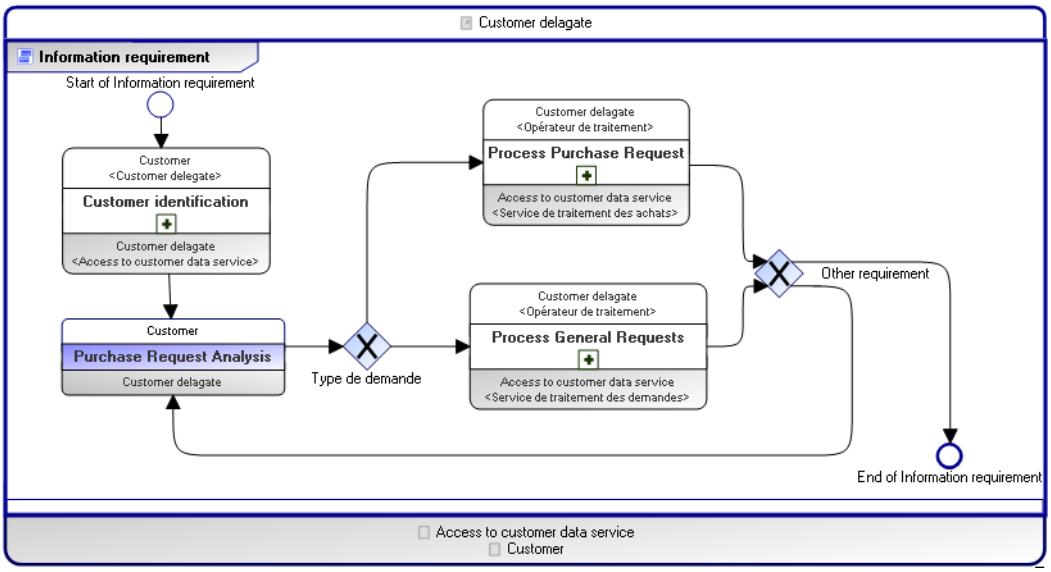


Diagramm eines Austauschvertrags (BPMN) "Antrag auf Informationen"

Der Austauschvertrag "Antrag auf Informationen" wird vom Call Center verwendet, um den Antrag eines Online-Kunden zu berücksichtigen. In diesem Austauschvertrag gibt es also drei verschiedene mitwirkende Personen: der Kunde, die IT-Anwendungen und der Kundenvertreter, der effektiv den Antrag des Dienstes stellt (hier das Call Center).

Beim Austauschvertrag wird der Kunde identifiziert und sein Antrag analysiert. Der Antrag wird im Anschluss als Einkaufsantrag oder als anderen Antrag bearbeitet, wenn es sich zum Beispiel um einen Antrag auf Auskunft handelt.

In den Diagrammen beschriebene Elemente

Die Diagramme der serviceorientierten Architekturen beruhen auf zwei Konzepttypen:

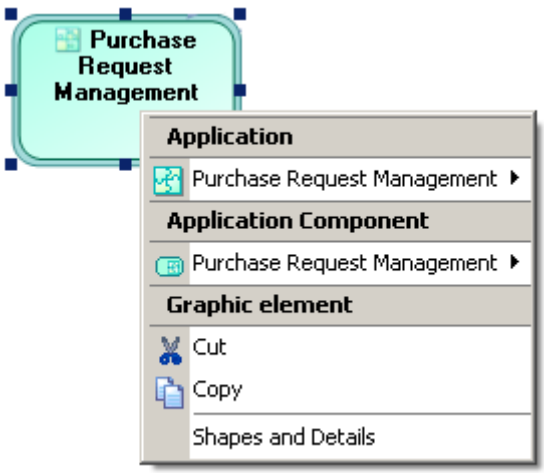
- den im Diagramm ***beschriebenen Elementen***, Hardware oder organisatorische Elemente, die in einer Architektur eingesetzt werden, wie zum Beispiel: die Verwendung eines Anwendungssystems, ein Endbenutzer oder eine Anwendungskomponente. Die mit diesen Elementen verbundenen Objekte können ohne Folgen für die Elemente selbst oder ihre Umgebung geändert werden.
- Die mit den beschriebenen Elementen ***verbundenen Objekte***, die konkreten Objekten entsprechen. Dabei handelt es sich um Anwendungsarchitekturen, Architekturen einer logischen Anwendung, Dienstvorgänge, OrgEinheiten oder Anwendungen.

  Im Beispiel der Architektur einer Bearbeitung der Einkaufsanträge sind "Plattform der Einkaufsverwaltung" und "Zahlungsmanagement" Verwendungen eines Anwendungssystems. Diese Anwendungssysteme setzen sich wiederum aus verschiedenen Ressourcen zusammen: Komponenten einer Anwendung und Verwendung eines Anwendungssystems.

Kontextmenü der beschriebenen Elemente

Bei allen Diagrammen in diesem Handbuch weist das Kontextmenü der meisten Komponenten (beispielsweise **Anwendungskomponente**) folgende Optionen auf:

- die Befehle des Objekttyps, der von dem Element benutzt wird (zum Beispiel **Anwendung**)
- die Befehle für die Komponente selbst
- die Befehle für die Grafik



Eigenschaften der beschriebenen Elemente

Um auf ein Objekt zuzugreifen, das die Komponente einer Architektur definiert, gehen Sie wie folgt vor:

1. Öffnen Sie das Eigenschaftsfenster der Komponente (zum Beispiel eine **Anwendungskomponente**) und wählen Sie die Registerkarte **Merkmale** aus.
   Das Objekt, das die Komponente definiert (hier eine **Anwendung**), wird im Feld **Verwendete Anwendung** angezeigt.
2. Klicken Sie auf den Pfeil rechts neben dem Feld **Verwendete Anwendung**, um das Kontextmenü des verwendeten Objekts zu öffnen oder das Objekt zu ersetzen, ohne die aktuelle Komponente zu ändern.

Diagramme und ihre Objekte

Die folgende Tabelle gibt einen Überblick über die Begriffe, die für die verschiedenen Elemente einer serviceorientierten Architektur verwendet werden und von dem Kontext abhängen, in dem die betroffene Komponente eingesetzt wird.

Diagramm	Diagrammobjekte	Verbundenes Objekt	Kommentar
Diagramm der Struktur eines Anwendungssystems (beschreibt ein Anwendungssystem)	Verwendung des Anwendungssystems	Anwendungssystem	Nutzung eines Anwendungssystems von einem anderen
	Anwendungskomponente	Anwendung	Verwendung einer Anwendung in einem Anwendungssystem
	Endbenutzer	OrgEinheit	
Diagramm einer Anwendungsstruktur (beschreibt eine Anwendung)	Anwendungskomponente	Anwendung	
	Komponente einer Plattformanwendung	Anwendung	Andere Form der Verwendung einer Anwendung in einer Architektur
Diagramm einer Architektur logischer Anwendungen (beschreibt logische Anwendungen)	Nutzung einer Architektur logischer Anwendungen	Architektur logischer Anwendungen	Nutzung der Architektur logischer Anwendungen von einer anderen
	Komponente einer logischen Anwendung	Logische Anwendung	Nutzung einer logischen Anwendung in einer Architektur logischer Anwendungen
	Endbenutzer	OrgEinheit	
Diagramm eines Austauschvertrags (BPMN) beschreibt einen Austauschvertrag	Verwendeter Austausch	Austausch	
	Verwendeter Austauschvertrag	Austauschvertrag	

EIGENSCHAFTEN DER VERBUNDENEN OBJEKTE

Sie können die Eigenschaften bestimmter Objekte angeben: eine Speichergröße für die Anwendungen oder eine Versionsnummer oder ein anderes besonderes Merkmal.

Diese Eigenschaften können für folgende Objekte definiert werden:

- ein Anwendungssystem
- eine Architektur logischer Anwendungen
- eine Organisationseinheit
- eine Anwendung

Eigenschaften erstellen

Um eine Eigenschaft zu erstellen, gehen Sie wie folgt vor:

1. Öffnen Sie das Eigenschaftsfenster des Objekts und klicken Sie auf die Registerkarte **Eigenschaften**.
2. Klicken Sie auf die Schaltfläche **Neu**.
   Daraufhin wird das Fenster zum Erstellen eines Eigenschaftswerts geöffnet.
3. Klicken Sie auf den Pfeil rechts neben dem Feld **Eigenschaftstyp** und wählen Sie anschließend **Erstellung von Eigenschaftstyp** aus.
   Daraufhin wird das Fenster zum Erstellen eines Eigenschaftstyps geöffnet.
4. Geben Sie den **Namen** an und klicken Sie auf **OK**.
5. Geben Sie bei den verschiedenen Objekten den **Wert** und die **Einheit** an.
6. Klicken Sie auf **OK**.
   Die Eigenschaft, der Wert und die Einheit werden in den Diagrammen angezeigt, die das Objekt beschreiben.

Vererbung der Eigenschaften

Wenn ein Objekt die Eigenschaften eines anderen Objekts erbt, wird der Wert und nicht der Typ der Eigenschaft vererbt.

```
Wenn Sie bei einem Objekt die Eigenschaft des Typs
"Versionsnummer" mit dem Wert "6.3" definiert haben, haben
alle vererbten Objekte dieses Objekts eine "Versionsnummer"
mit dem Wert "6.3".
```

Um den Wert einer vererbten Eigenschaft zu ändern, gehen Sie wie folgt vor:

1. Erstellen Sie eine neue Eigenschaft desselben Typs mit einem neuen Wert (zum Beispiel die "Versionsnummer der XP-Arbeitsplätze")

2. Ersetzen Sie die vererbte Eigenschaft durch eine neue.

☛ *Weitere Informationen zu den Variationen finden Sie im Handbuch **MEGA Common Features**, Kapitel "Objekte des Repositorys bearbeiten", "Objektvariationen".*

MIT MEGA SYSTEM ORIENTED IT ARCHITECTURE MODELLIEREN

MEGA System Oriented IT Architecture ermöglicht, die Architektur der Anwendungen in Ihrem Unternehmen nach einer serviceorientierten Methode mit zwei Konzepten zu beschreiben:

- die Anwendungssysteme
- die Architekturen logischer Anwendungen

Hier werden folgende Punkte erörtert:

✓ "Anwendungssystem beschreiben", Seite 38
✓ "Architektur logischer Anwendungen beschreiben", Seite 45
✓ "Data Stores verwenden", Seite 48
✓ "Dienst- und Abfragepunkte beschreiben", Seite 52
✓ "Interaktionen verwalten", Seite 55

ANWENDUNGSSYSTEM BESCHREIBEN

Ein *Anwendungssystem* wird von dem Diagramm einer Struktur eines Anwendungssystems beschrieben.

> 🕮 *Ein Anwendungssystem ist eine Montage aus anderen Anwendungssystemen, Anwendungen und Endbenutzern, die mit den Anwendungskomponenten zusammenwirken, um eine oder mehrere Funktionalitäten durchzuführen.*

Das Anwendungssystem zeichnet sich aus durch:

- angebotene Dienste: Diese Dienste werden durch Dienstpunkte dargestellt.
- verwendete Dienste: Diese Dienste werden durch Abfragepunkte dargestellt.

> ☛ *Weitere Informationen finden Sie unter "Dienst- und Abfragepunkte beschreiben", Seite 52.*

Anwendungssystem erstellen

Um ein *Anwendungssystem* ab einer Bibliothek zu erstellen, gehen Sie wie folgt vor:

1. Wählen Sie die gewünschte Bibliothek aus.
2. Klicken Sie mit der rechten Maustaste auf den Ordner "Anwendungssystem" und klicken Sie anschließend auf **Neu > Anwendungssystem**.
   Daraufhin wird das Fenster **Erstellung von Anwendungssystem** geöffnet.
3. Geben Sie den **Namen** des Anwendungssystems ein und klicken Sie auf **OK**.
   Das neue Anwendungssystem wird in der Navigationsbaumstruktur angezeigt.

Komponenten eines Anwendungssystems beschreiben

Das Diagramm der Struktur eines Anwendungssystems beschreibt die interne Struktur eines Anwendungssystems:

- die angebotenen oder erforderlichen Dienste
- die übernommenen Prozesse, die Komponenten und ihre Interaktionen
- die Endbenutzer in Interaktion mit den Anwendungskomponenten

```
Folgendes Diagramm beschreibt das Anwendungssystem für die
Bearbeitung der Einkaufsanträge.
```

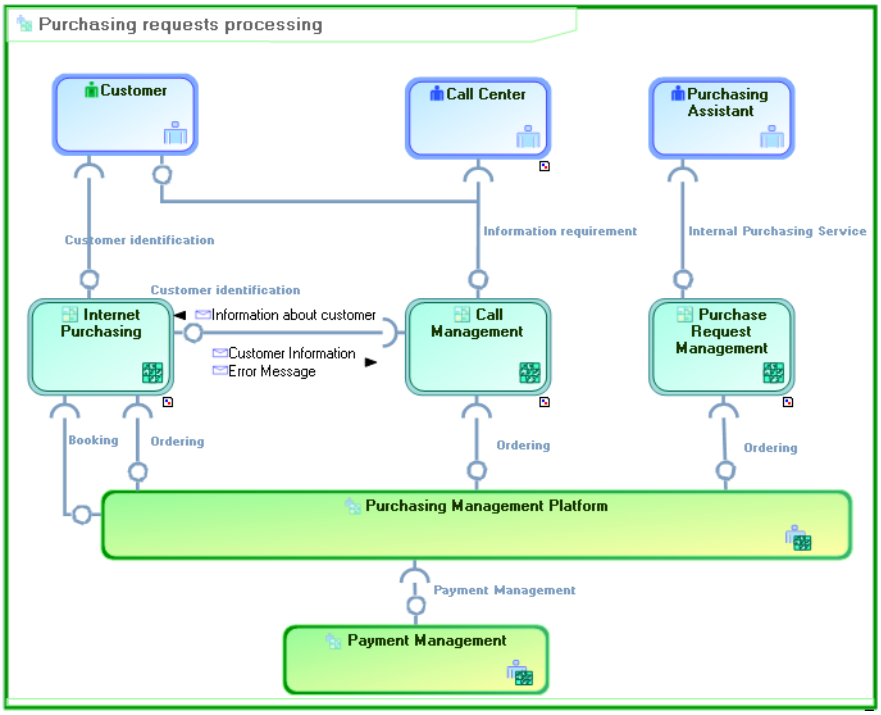


Diagramm der Struktur des Anwendungssystems "Bearbeitung der Einkaufsanträge"

Um das Diagramm der Struktur eines Anwendungssystems mit **Windows Front-End**zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie mit der rechten Maustaste auf das Anwendungssystem und klicken Sie auf **Neu > Diagramm**.
2. Wählen Sie in dem nun angezeigten Fenster "Strukturdiagramm eines Anwendungssystems" aus und überprüfen Sie, ob das Kästchen **Diagramm initialisieren** markiert ist. Klicken Sie anschließend auf **Erstellen**.
   Daraufhin wird das Diagramm geöffnet. Sie befinden sich im Grafik-Editor von **MEGA**.

Um das Diagramm der Struktur eines Anwendungssystems mit **Web Front-End**zu erstellen, gehen Sie wie folgt vor:

- Klicken Sie mit der rechten Maustaste auf das Anwendungssystem und klicken Sie auf **Neu > Strukturdiagramm eines Anwendungssystems**.

Verwendung eines Anwendungssystems erstellen

Um zu beschreiben, dass ein Anwendungssystem ein anderes Anwendungssystem verwendet, gehen Sie wie folgt vor:

- Erstellen Sie eine Komponente des Typs *Verwendung eines Anwendungssystems*
- Verbinden Sie die verwendete Architektur mit der Komponente.

Die Nutzung einer Architektur logischer Anwendungen beschreibt die Rolle einer Architektur logischer Anwendungen in einer zusammengesetzten Beziehung mit einer übergeordneten Architektur logischer Anwendungen.

```
Bei der Architektur der Bearbeitung von Einkaufsanträgen
zum Beispiel kommen die Dienste der Anwendungsarchitekturen
der "Plattform der Einkaufsverwaltung" und
"Zahlungsmanagement" zum Einsatz.
```

Um eine **Verwendung eines Anwendungssystems** zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie in der Objektleiste des Diagramms der Struktur eines Anwendungssystems auf die Schaltfläche **Verwendung des Anwendungssystems**.
2. Klicken Sie auf den Rahmen des beschriebenen Anwendungssystems. In einem Fenster werden Sie aufgefordert, das **Verwendete Anwendungssystem** auszuwählen. Es handelt sich um das genutzte Anwendungssystem (beispielsweise "Zahlungsmanagement").
3. Wählen Sie ein existierendes Anwendungssystem aus oder erstellen Sie ein neues.
4. Klicken Sie auf **Beenden**.
   Die Verwendung des Anwendungssystems wird im Diagramm angezeigt.

Anwendungskomponente erstellen

Um zu beschreiben, dass ein Anwendungssystem eine Anwendung verwendet, gehen Sie wie folgt vor:

- Erstellen Sie eine *Anwendungskomponente*
- Verknüpfen Sie mit dieser Komponente die verwendete *Anwendung*

Eine IT-Service-Komponente stellt die Tatsache dar, dass der verwendete IT-Service eine Rolle in der beschriebenen Anwendung spielt.

Ein lokales Data Store ist ein Data Store, das ausschließlich innerhalb des beschriebenen Systems verwendet wird.

Um eine **Anwendungskomponente** zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie in der Objektleiste des Diagramms der Struktur eines Anwendungssystems auf die Schaltfläche **Anwendungskomponente** und anschließend in den Rahmen des beschriebenen Anwendungssystems.
   In einem Fenster werden Sie aufgefordert, die **Verwendete Anwendung** auszuwählen.
2. Wählen Sie eine existierende Anwendung aus oder erstellen Sie eine neue Anwendung.
3. Klicken Sie auf **OK**.
   Die Anwendungskomponente wird im Diagramm angezeigt.

Endbenutzer erstellen

Um zu beschreiben, dass ein Anwendungssystem wie die Bearbeitung der Einkaufsanträge von internen und externen OrgEinheiten aktiviert wurde, gehen Sie wie folgt vor:

- Erstellen Sie eine Komponente des Typs *Endbenutzer*
- Verknüpfen Sie mit dieser Komponente eine OrgEinheit, zum Beispiel "Assistent Einkauf".

> *Ein Endbenutzer stellt eine OrgEinheit dar, die an den Grenzen eines Anwendungssystems oder einer Architektur logischer Anwendungen mitwirkt.*

Um einen Endbenutzer zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie in der Objektleiste des Diagramms der Struktur eines Anwendungssystems auf die Schaltfläche **Endbenutzer** und anschließend in den Rahmen des Diagramms des Anwendungssystems.
   In einem Fenster werden Sie aufgefordert, die eingebundene organisatorische Ressource auszuwählen.
2. Wählen Sie die betreffende OrgEinheit aus und klicken Sie auf **OK**.

Beschreibung eines Anwendungssystems ergänzen

Zugriff auf die Beschreibung eines Anwendungssystems

Im Eigenschaftsfenster des Anwendungssystems können Sie auf die vollständige Beschreibung dieses Anwendungssystems zugreifen.

Um auf die Beschreibung eines Anwendungssystems zuzugreifen, gehen Sie wie folgt vor:

1. Klicken Sie mit der rechten Maustaste auf das betreffende Anwendungssystem und wählen Sie **Eigenschaften** aus.
   Die Informationen zum Anwendungssystem werden in verschiedenen Registerkarten aufgeführt.

Um auf alle Objekte des Systems zuzugreifen, gehen Sie wie folgt vor:

❱ Klicken Sie auf die Registerkarte **Komponenten**.

☛ *Weitere Informationen finden Sie unter "Komponenten eines Anwendungssystems beschreiben", Seite 39.*

Um auf die Eigenschaften zuzugreifen, die in den Diagrammen eines Anwendungssystems im unteren Teil des Rahmens des beschriebenen Anwendungssystems aufgeführt werden, gehen Sie wie folgt vor:

❱ Klicken Sie auf die Registerkarte **Eigenschaften**.

☛ *Weitere Informationen finden Sie unter "Eigenschaften der verbundenen Objekte", Seite 35.*

Um auf die Liste der von der Architektur durchgeführten oder verwendeten Dienste zuzugreifen, gehen Sie wie folgt vor:

❱ Klicken Sie auf die Registerkarte **Dienst- und Abfragepunkte**.

☛ *Weitere Informationen finden Sie unter "Dienst- und Abfragepunkte beschreiben", Seite 52.*

Anwendungssystem mit einem Systemprozess verbinden

Ein Anwendungssystem kann definiert werden, um einen *Systemprozess* auszuführen.

🕮 *Ein Systemprozess ist die ausführbare Darstellung eines Prozesses. Folgende Elemente formalisieren einen Systemprozess: Workflow-Elemente, zu erfüllende Tasks während der Bearbeitung, algorithmische Elemente eines Sequenzflusses von Tasks, mit Teilnehmern ausgetauschte Informationsflüsse.*

Um ein Anwendungssystem mit einem Systemprozess zu verbinden, gehen Sie wie folgt vor:

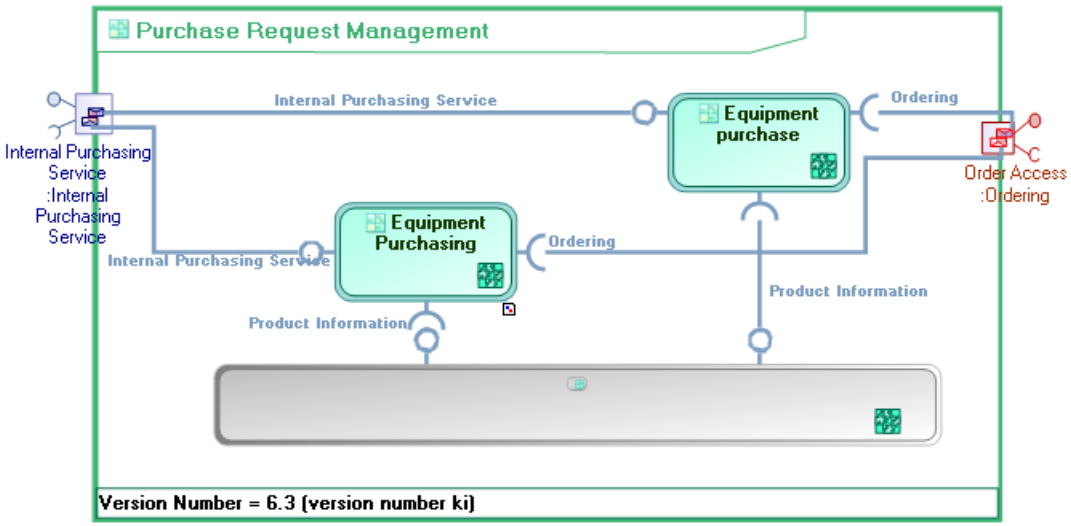
1. Klicken Sie mit der rechten Maustaste auf das Anwendungssystem und wählen Sie **Neu > Prozessdurchführung** aus.
   Daraufhin wird das Fenster **Erstellung von Verwendung einer Anwendung durch einen Prozess** angezeigt.
2. Wählen Sie im Feld **Ausgeführter Prozess** den betreffenden Systemprozess aus und klicken Sie auf **OK**.
   Der Systemprozess wird im Eigenschaftsfenster des Anwendungssystems in der Registerkarte **Ausführung** angezeigt.

Struktur einer Anwendung beschreiben

Mit **MEGA System Oriented IT Architecture** kann eine Anwendung mit einem Diagramm der Anwendungsstruktur beschrieben werden.

🕮 *In einem Diagramm der Anwendungsstruktur werden die Komponenten auf der ersten Ebene einer Anwendung, die*

Zugriffspunkte (Dienst- und Abfragepunkt) und die Verbindungen zwischen den Komponenten grafisch dargestellt.



Die Anwendung zur Verwaltung der Einkaufsanträge wird nicht für interne Einkäufe verwendet und stützt sich auf zwei spezielle Anwendungen: eine für den Einkauf von Gebrauchsartikeln und eine andere für die Materialbeschaffung. Die beiden Anwendungen verwenden Microsoft Excel.

☛ *Die Versionsnummer im unteren Bereich des Rahmens ist eine Eigenschaft der Anwendung. Weitere Informationen finden Sie unter "Eigenschaften der beschriebenen Elemente", Seite 33.*

Das Diagramm besteht aus:

- *Anwendungskomponenten*, die die verwendeten Anwendungen darstellen

  🕮 *Eine IT-Service-Komponente stellt die Tatsache dar, dass der verwendete IT-Service eine Rolle in der beschriebenen Anwendung spielt.*

  Im Beispiel wird die Anwendung für den Einkauf von Gebrauchsartikeln und die Anwendung für die Materialbeschaffung aufgeführt.

- *Komponenten einer Plattformanwendung*, die die von der Plattform verwendeten Technologien darstellen

  Im Beispiel wird die Anwendung Microsoft Excel aufgeführt.

  🕮 *Komponente einer Plattformanwendung stellt die Tatsache dar, dass die verwendete Technologie in der beschriebenen Anwendung die*

Rolle einer Plattform übernimmt. Beispielsweise ist Tomcat eine Plattformkomponente für SageCRM.

📖 *Eine Technologie ist eine Definition oder ein Format, das von einer Normungsorganisation zugelassen wurde oder von der Industrie als Standard akzeptiert wird.*

- *IT-Service-Komponente*, die die verwendeten IT-Services darstellen

  `Diese Komponenten werden im Beispiel nicht verwendet.`

  📖 *Eine IT-Service-Komponente stellt die Tatsache dar, dass der verwendete IT-Service eine Rolle in der beschriebenen Anwendung spielt.*

  📖 *Ein IT-Service ist das Teilelement einer Anwendung, das dem Endbenutzer dieser Anwendung im Rahmen seiner Arbeit zur Verfügung gestellt wird.*

- Zugriffs-, Abfrage- und Dienstpunkte

  ☛ *Weitere Informationen finden Sie unter "Dienst- und Abfragepunkte beschreiben", Seite 52.*

- *Interaktionen* zwischen den Komponenten

  📖 *Eine Interaktion ist ein Vertrag, der in einem bestimmten Kontext zwischen zwei autonomen Entitäten innerhalb oder außerhalb eines Unternehmens abgeschlossen wird. Diese Entitäten können Organisationseinheiten, Aktivitäten oder Prozesse des Unternehmens sein, sowie externe Orgeinheiten eines Unternehmens. Der Inhalt dieses Vertrages wird von einem Austauschvertrag beschrieben.*

ARCHITEKTUR LOGISCHER ANWENDUNGEN BESCHREIBEN

Eine *logische Anwendung* ist ein Element einer IT-Architektur.

> 🕮 *Eine logische Anwendung ist das Teilelement einer Anwendung, das dem Endbenutzer dieser Anwendung im Rahmen seiner Arbeit zur Verfügung gestellt wird. Die logischen Anwendungen stellen die Aufteilung der Anwendungen in Form funktionaler elementarer Einheiten dar. Eine logische Anwendung ist eine kohärente und nicht teilbare Bearbeitungseinheit, die eine Gruppe von Nachrichten und Ereignissen koordiniert, um im Informationssystem eine Task durchzuführen. Im Anwendungssystem entspricht die logische Anwendung einer funktionellen Einheit, die weitaus elementarer ist.*

Eine *Architektur logischer Anwendungen* ist eine Komponente, mit der die Organisation *logischer Anwendungen* nach einer serviceorientierten Methode beschrieben wird.

> 🕮 *Die Architektur logischer Anwendungen ist eine Montage aus anderen Architekturen, logischen Anwendungen und Endbenutzern, die mit den Anwendungskomponenten zusammenwirken, um eine oder mehrere Funktionalitäten durchzuführen.*
>
> 💣 **In MEGA System Oriented IT Architecture gibt es keine direkte Verbindung zwischen einem Anwendungssystem und der Architektur logischer Anwendungen.**

Architektur logischer Anwendungen erstellen

Um eine *Architektur logischer Anwendungen* zu erstellen, gehen Sie wie folgt vor:

1. Wählen Sie die gewünschte Bibliothek aus.
2. Klicken Sie mit der rechten Maustaste auf den Ordner "Logische Anwendungen" und wählen Sie anschließend **Neu > Architektur logischer Anwendungen** aus.
   Das Fenster **Erstellung von Architektur logischer Anwendungen** wird angezeigt.
3. Geben Sie den **Namen** der Architektur der logischen Anwendungen ein und klicken Sie auf **OK**.
   Die neue Architektur logischer Anwendungen wird in der Navigationsbaumstruktur aufgeführt.

Komponenten der Architektur logischer Anwendungen beschreiben

Ein Diagramm der Architektur logischer Anwendungen beschreibt die interne Struktur der Architektur:

- die angebotenen oder erforderlichen Dienste
- die übernommenen Prozesse, die Komponenten und ihre Interaktionen
- die Endbenutzer in Interaktion mit den Softwarekomponenten

Folgendes Diagramm beschreibt die Architektur der logischen Anwendung, die den Warenkorb des Kunden verwaltet.

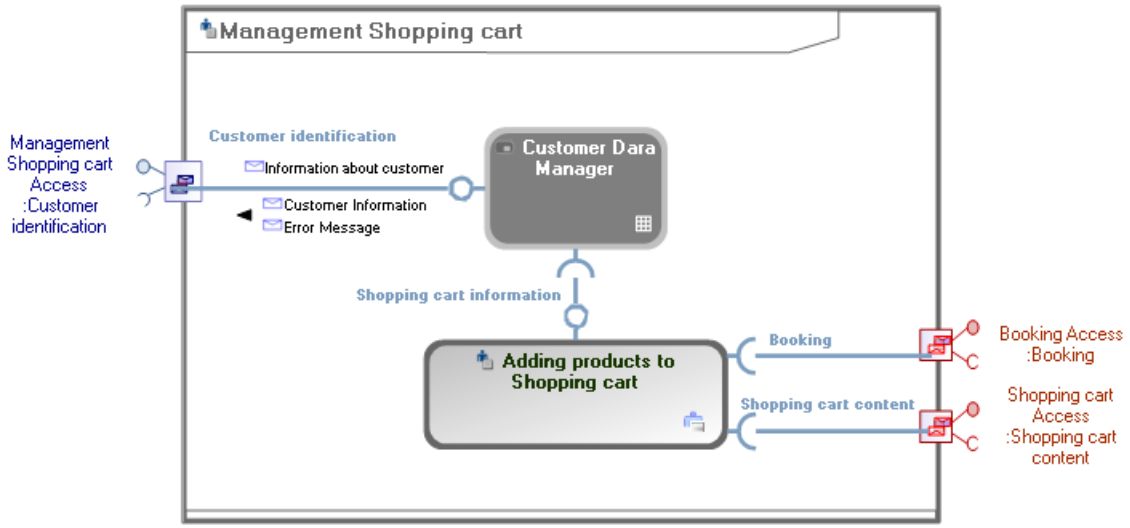


Diagramm einer Architektur logischer Anwendungen "Warenkorb des Kunden verwalten"

Ein Diagramm der Architektur logischer Anwendungen wird nach demselben Prinzip erstellt wie ein Diagramm der Struktur eines Anwendungssystems.

☛ *Weitere Informationen zur Erstellung von Komponenten dieses Diagramms finden Sie unter "Komponenten eines Anwendungssystems beschreiben", Seite 39.*

Ein Diagramm der Architektur logischer Anwendungen setzt sich zusammen aus:

- *Endbenutzern*

  📖 *Ein Endbenutzer stellt eine OrgEinheit dar, die an den Grenzen eines Anwendungssystems oder einer Architektur logischer Anwendungen mitwirkt.*

- *Verwendungen einer Architektur logischer Anwendungen*

  📖 *Die Nutzung einer Architektur logischer Anwendungen beschreibt die Rolle einer Architektur logischer Anwendungen in einer*

zusammengesetzten Beziehung mit einer übergeordneten Architektur logischer Anwendungen.

- *Komponenten einer logischen Anwendung*

  Die Komponente einer logischen Anwendung beschreibt die Rolle einer logischen Anwendung in einer zusammengesetzten Beziehung mit einer übergeordneten logischen Anwendung oder einer übergeordneten Architektur logischer Anwendungen.

- *Interaktionen* zwischen den Komponenten, die die Dienstanträge darstellen

  Eine Interaktion ist ein Vertrag, der in einem bestimmten Kontext zwischen zwei autonomen Entitäten innerhalb oder außerhalb eines Unternehmens abgeschlossen wird. Diese Entitäten können Organisationseinheiten, Aktivitäten oder Prozesse des Unternehmens sein, sowie externe Orgeinheiten eines Unternehmens. Der Inhalt dieses Vertrages wird von einem Austauschvertrag beschrieben.

  ☛ *Weitere Informationen zu den Interaktionen zwischen den Komponenten der Architektur logischer Anwendungen finden Sie unter "Interaktionen verwalten", Seite 55.*

- *Dienstpunkten*

  Ein Dienstpunkt ist ein Austauschpunkt, über den ein Agent potenziellen Kunden einen Dienst anbietet.

- *Abfragepunkten*

  Ein Abfragepunkt ist ein Austauschpunkt, über den ein Agent bei potenziellen Lieferanten einen Dienst beantragt.

  ☛ *Weitere Informationen zu den Zugriffspunkten finden Sie unter "Dienst- und Abfragepunkte beschreiben", Seite 52.*

DATA STORES VERWENDEN

> 🕮 *Ein Data Store bietet einen Mechanismus an, mit dem Daten aktualisiert oder abgerufen werden, die auch über die Reichweite des aktuellen Prozesses hinaus bestehen bleiben. In einem Data Store werden die Daten der Flüsse am Eingang gespeichert und über einen oder mehrere Flüsse am Ausgang weitergeleitet.*

Data Stores können in folgenden Diagrammen beschrieben werden:

- Strukturdiagramme eines Anwendungssystems
- Diagramme über Anwendungsstruktur
- Diagramme der Architektur einer logischen Anwendung

Physische Data Stores beschreiben

> 🕮 *Ein physisches Data Store stellt die Implementierung eines logischen Data Stores dar.*

Im Diagramm der Struktur eines Anwendungssystems und im Diagramm der Anwendungsstruktur können Sie nur folgende physische Data Stores verwenden:

- Lokale oder externe
- RE (Relation Entität) oder RDB (Relationale Datenbank).

> 🕮 *Ein Relationale Datenbank-Data Store ist ein Data Store in Form von Tabellen oder Tabellenansichten.*

> 🕮 *Ein Entität/Relation-Data Store beschreibt die Daten einer Datenbank in Form von Entitäten oder Klassen.*

Physisches lokales RDB-Data Store erstellen

> 🕮 *Ein lokales Data Store ist ein Data Store, das ausschließlich innerhalb des beschriebenen Systems verwendet wird.*

Um zum Beispiel ein physisches lokales RDB-Data Store ab einem Diagramm der Struktur eines Anwendungssystems zu erstellen, gehen Sie wie folgt vor:

1. Öffnen Sie das betreffende Diagramm.
2. Klicken Sie in der Objektleiste des Diagramms der Struktur eines Anwendungssystems auf die Schaltfläche **Physisches lokales Data Store RDB**.
3. Klicken Sie auf den Rahmen des beschriebenen Anwendungssystems.
   In einem Fenster werden Sie aufgefordert, die **Struktur des Datenbestands** auszuwählen. Es handelt sich um die physische RDB-Struktur, die das Data Store konkret unterstützt.
4. Wählen Sie eine existierende **Struktur des Datenbestands** aus oder erstellen Sie eine neue.
5. Klicken Sie auf **Beenden**.
   Das physische lokale RDB-Data Store wird im Diagramm angezeigt und trägt den Namen der Struktur des Datenbestands.

> ☛ *Die Struktur eines RDB-Datenbestands wird von einem Paket gehalten.*

Physisches externes RE-Data Store erstellen

> 🕮 *Ein externes Data Store ist ein Data Store, das innerhalb und außerhalb des beschriebenen Systems verwendet wird.*

Um zum Beispiel ein physisches externes RE-Data Store ab einem Diagramm der Struktur eines Anwendungssystems zu erstellen, gehen Sie wie folgt vor:

1. Öffnen Sie das betreffende Diagramm.
2. Klicken Sie in der Objektleiste des Diagramms der Struktur eines Anwendungssystems auf die Schaltfläche **Physisches externes RE-Data Store erstellen**.
3. Klicken Sie auf den Rahmen des beschriebenen Anwendungssystems.
   In einem Fenster werden Sie aufgefordert, die **Struktur des Datenbestands** auszuwählen. Es handelt sich um die physische ER-Struktur, die das Data Store konkret unterstützt.
4. Wählen Sie eine existierende **Struktur des Datenbestands** aus oder erstellen Sie eine neue.
5. Klicken Sie auf **Beenden**.
   Das physische externe RE-Data Store wird am Rand des Diagramms angezeigt und trägt den Namen der Struktur des Datenbestands.

> ☛ *Die Struktur eines RE-Datenbestands wird von einer Datenbank gehalten.*

Zugriff auf ein physisches Data Store beschreiben

Folgende Komponenten können im Lese- oder Schreibmodus auf ein physisches Data Store zugreifen.

- Die Elemente vom Typ *Verwendung des Anwendungssystems*, die die beiden verwendeten Anwendungssysteme darstellen.

  > 🕮 *Die Nutzung eines Anwendungssystems beschreibt die Rolle eines Anwendungssystems in einer zusammengesetzten Beziehung mit einem übergeordneten Anwendungssystem.*

- Die *Anwendungskomponenten*, die die verwendeten Anwendungen darstellen

  > 🕮 *Eine IT-Service-Komponente stellt die Tatsache dar, dass der verwendete IT-Service eine Rolle in der beschriebenen Anwendung spielt.*

- Die *Komponenten einer Plattformanwendung*.

  > 🕮 *Komponente einer Plattformanwendung stellt die Tatsache dar, dass die verwendete Technologie in der beschriebenen Anwendung die Rolle einer Plattform übernimmt. Beispielsweise ist Tomcat eine Plattformkomponente für SageCRM.*

Um den Zugriff auf das Data Store im Lesemodus zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie in der Objektleiste des Diagramms auf die Schaltfläche **Verbindung**.
2. Ziehen Sie eine Verbindung zwischen dem Data Store und der Entität, die die Daten liest (Komponente oder Verwendung eines Anwendungssystems).
   Eine Verbindung vom Data Store zur Entität wird erstellt.

> ☛ *Um eine Verbindung im Schreibmodus zu erstellen, müssen Sie eine Verbindung zwischen der schreibenden Entität und dem Data Store erstellen.*

Logisches Data Store beschreiben

> 🕮 *Ein logisches Data Store stellt die Verwendung der Daten über Anwendungssysteme dar, ohne zu berücksichtigen, wie der Zugriff konkret implementiert wird.*

Im Diagramm einer Architektur der logischen Anwendung können Sie nur logische lokale oder externe RE-Data Stores verwenden.

> 🕮 *Die Architektur logischer Anwendungen ist eine Montage aus anderen Architekturen, logischen Anwendungen und Endbenutzern, die mit den Anwendungskomponenten zusammenwirken, um eine oder mehrere Funktionalitäten durchzuführen.*

> 🕮 *Eine logische Anwendung ist das Teilelement einer Anwendung, das dem Endbenutzer dieser Anwendung im Rahmen seiner Arbeit zur Verfügung gestellt wird. Die logischen Anwendungen stellen die Aufteilung der Anwendungen in Form funktionaler elementarer Einheiten dar. Eine logische Anwendung ist eine kohärente und nicht teilbare Bearbeitungseinheit, die eine Gruppe von Nachrichten und Ereignissen koordiniert, um im Informationssystem eine Task durchzuführen. Im Anwendungssystem entspricht die logische Anwendung einer funktionellen Einheit, die weitaus elementarer ist.*

Logisches lokales Data Store erstellen

> 🕮 *Ein lokales Data Store ist ein Data Store, das ausschließlich innerhalb des beschriebenen Systems verwendet wird.*

Um zum Beispiel ein logisches lokales Data Store ab einem Diagramm einer Architektur der logischen Anwendung zu erstellen, gehen Sie wie folgt vor:

1. Öffnen Sie das betreffende Diagramm.
2. Klicken Sie in der Objektleiste des Diagramms der Struktur eines Anwendungssystems auf die Schaltfläche **Logisches lokales Data Store erstellen**.
3. Klicken Sie in den Rahmen der beschriebenen Anwendungsarchitektur. In einem Fenster werden Sie aufgefordert, die **Struktur des logischen Data Store** auszuwählen. Es handelt sich um die physische ER-Struktur, die das Data Store konkret unterstützt.
4. Wählen Sie eine existierende **Struktur des logischen Data Store** aus oder erstellen Sie eine neue.
5. Klicken Sie auf **Beenden**.
   Das logische lokale Data Store wird im Diagramm angezeigt und trägt den Namen der Struktur des logischen RE-Data Stores.

   > ☛ *Die Struktur eines logischen RE-Data Stores wird von einem Paket gehalten.*

Zugriff auf ein logisches Data Store beschreiben

Folgende Komponenten können im Lese- oder Schreibmodus auf ein logisches Data Store zugreifen.

- Die *Verwendungen einer Architektur logischer Anwendungen*

  > 🕮 *Die Nutzung einer Architektur logischer Anwendungen beschreibt die Rolle einer Architektur logischer Anwendungen in einer*

zusammengesetzten Beziehung mit einer übergeordneten Architektur logischer Anwendungen.

- Die *Komponenten einer logischen Anwendung*, die die verwendeten logischen Anwendungen im Kontext der beschriebenen Architektur einer logischen Anwendung darstellen.

  Die Komponente einer logischen Anwendung beschreibt die Rolle einer logischen Anwendung in einer zusammengesetzten Beziehung mit einer übergeordneten logischen Anwendung oder einer übergeordneten Architektur logischer Anwendungen.

Um den Zugriff auf das logische Data Store im Lesemodus zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie in der Objektleiste des Diagramms auf die Schaltfläche **Verbindung**.
2. Ziehen Sie eine Verbindung zwischen dem Data Store und der Entität, die die Daten liest (Komponente einer logischen Anwendung oder Verwendung der Architektur logischer Anwendungen).
   Eine Verbindung vom Data Store zur Entität wird erstellt.

   Um eine Verbindung im Schreibmodus zu erstellen, müssen Sie eine Verbindung zwischen der schreibenden Entität und dem Data Store erstellen.

Realisierung eines Data Stores

Um alle Elemente der Anwendungsstruktur zu beschreiben, die von einem Data Store implementiert wird, definieren Sie bei den betreffenden Data Stores eine *Realisierung*.

Eine Realisierung beschreibt die Beziehung zwischen einer logischen und einer physischen Entität, mit der sie umgesetzt wird. Die physische Entität ergibt eine Liste der logischen Entitäten, die von ihr implementiert werden.

Um zu beschreiben, dass ein Data Store ein Element der Anwendungsstruktur implementiert, gehen Sie wie folgt vor:

1. Öffnen Sie das Eigenschaftsfenster des betreffenden Data Stores.
2. Wählen Sie die Registerkarte **Merkmale** und die Unterregisterkarte **Realisierung** aus.
3. Klicken Sie im Abschnitt **Realisierung einer Komponente** auf die Schaltfläche **Neu**.
   Das Fenster zur Erstellung der Realisierung eines Data Stores wird angezeigt.
4. Geben Sie im Feld **Durchführungskontext**, die logische Anwendung oder die Architektur der logischen Anwendung an.
5. Wählen Sie im Feld **Realisierte Komponente** unter den Komponenten die zu implementierende Komponente aus.
6. Klicken Sie auf **OK**.
   Die Realisierung des Data Stores wird auf der Eigenschaftsseite angezeigt.

DIENST- UND ABFRAGEPUNKTE BESCHREIBEN

In einer serviceorientierten Architektur verwenden die Kommunikationen:

- *Interaktionen*

  Eine Interaktion ist ein Vertrag, der in einem bestimmten Kontext zwischen zwei autonomen Entitäten innerhalb oder außerhalb eines Unternehmens abgeschlossen wird. Diese Entitäten können Organisationseinheiten, Aktivitäten oder Prozesse des Unternehmens sein, sowie externe Orgeinheiten eines Unternehmens. Der Inhalt dieses Vertrages wird von einem Austauschvertrag beschrieben.

  ☛ *Weitere Informationen zu den Interaktionen zwischen den Komponenten der Architektur logischer Anwendungen finden Sie unter "Interaktionen verwalten", Seite 55.*

- Zugriffpunkte: *Dienst-* und *Abfragepunkte*

  Ein Abfragepunkt ist ein Austauschpunkt, über den ein Agent bei potenziellen Lieferanten einen Dienst beantragt.

  Ein Dienstpunkt ist ein Austauschpunkt, über den ein Agent potenziellen Kunden einen Dienst anbietet.

MEGA System Oriented IT Architecture ermöglicht, die einzelnen Komponenten zu verwalten und zu überprüfen, ob die angegebenen Informationen in sich übereinstimmen.

Konzepte - Einführung

Dienstpunkte

Ein Anwendungssystem oder eine Architektur logischer Anwendungen wird erstellt, um einen oder mehrere Dienste sicherzustellen. Diese Dienste werden mit *Dienstpunkten* dargestellt.

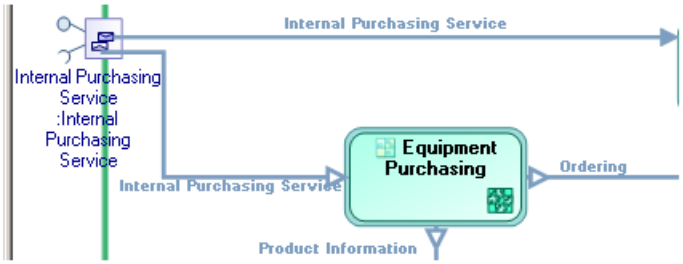
Ein Dienstpunkt ist ein Austauschpunkt, über den ein Agent potenziellen Kunden einen Dienst anbietet.

Der Dienst wird nach ganz genauen Modalitäten beantragt, die durch einen *Austauschvertrag* definiert werden, der dem Dienstpunkt zugewiesen ist.

Ein Austauschvertrag ist ein Vertragsmodell zwischen zwei Organisationseinheiten. Dieser Vertrag wird von den Austauschen zwischen der Rolle eines Initiators und einer oder mehreren Rollen der Beitragsleistenden beschrieben.

Die für einen Dienst aktivierten Komponenten sind über Interaktionen mit dem Dienstpunkt verbunden. Sollte es notwendig sein, mehrere Komponenten zu

aktivieren, müssen Sie mehrere Interaktionen zwischen dem Dienstpunkt und den Komponenten der Architektur erstellen.



In dem hier aufgeführten Beispiel ist der Dienste Interner Einkauf mit zwei Interaktionen verbunden, die denselben Austauschvertrag verwenden und die Aktivierung der Anwendung Materialbeschaffung oder der Anwendung Einkauf von Gebrauchsartikeln darstellen.

☛ *Informationen zum Erstellen eines Dienstpunkts finden Sie unter* *"Dienstpunkt oder Abfragepunkt erstellen", Seite 54*.

Abfragepunkte

Mit einem *Abfragepunkt* wird die Verwendung eines externen Dienstes dargestellt.

🕮 *Ein Abfragepunkt ist ein Austauschpunkt, über den ein Agent bei potenziellen Lieferanten einen Dienst beantragt.*

Der Dienst wird nach ganz genauen Modalitäten beantragt, die durch einen *Austauschvertrag* definiert werden, der dem Abfragepunkt zugewiesen ist.

🕮 *Ein Austauschvertrag ist ein Vertragsmodell zwischen zwei Organisationseinheiten. Dieser Vertrag wird von den Austauschen zwischen der Rolle eines Initiators und einer oder mehreren Rollen der Beitragsleistenden beschrieben.*

Die Komponenten, die eine Abfrage senden, sind mit dem Abfragepunkt über eine Interaktion verbunden.

In dem Beispiel stellen die Abfragepunkte die Dienstanträge der Verwendung einer logischen Anwendung "Produkte in Warenkorb hinzufügen" dar, um eine Bestellung aufzugeben oder Produkte zu reservieren.

☛ *Informationen zum Erstellen eines Abfragepunkts finden Sie unter* *"Dienstpunkt oder Abfragepunkt erstellen", Seite 54*.

Dienstpunkt oder Abfragepunkt erstellen

Die Erstellung eines *Dienstpunktes* ist mit der eines *Abfragepunktes* identisch.

> 🕮 *Ein Abfragepunkt ist ein Austauschpunkt, über den ein Agent bei potenziellen Lieferanten einen Dienst beantragt.*
>
> 🕮 *Ein Dienstpunkt ist ein Austauschpunkt, über den ein Agent potenziellen Kunden einen Dienst anbietet.*

Um einen Dienstpunkt zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie in der Objektleiste des Diagramms auf die Schaltfläche **Dienstpunkt** .
2. Legen Sie das Objekt am Rand des Rahmens der Architektur ab.
   Dann wird das Fenster zum Erstellen eines Dienstpunkts angezeigt.
3. Klicken Sie auf den Pfeil rechts neben dem Feld **Austauschvertrag**, um den Austauschvertrag zur Aktivierung dieses Dienstpunktes zu definieren, und wählen Sie zum Beispiel **Austauschvertrag definieren** aus.
   Daraufhin wird das Abfragefenster angezeigt.
4. Wählen Sie den Austauschvertrag aus, der mit dem Dienstpunkt verknüpft ist.
5. Klicken Sie auf **Weiter**.
   In einem Fenster wird nun die Liste der Rollen des Austauschvertrags angezeigt, die mit dem Dienstpunkt verbunden werden können.

   > ☛ *Dieses neue Fenster wird nicht angezeigt, wenn mit dem Dienstpunkt nur eine einzige Kandidatenrolle verbunden werden soll.*
6. Wählen Sie die betreffende Rolle aus und klicken Sie auf **OK**.
   Der Dienstpunkt wird in dem Diagramm angezeigt.

Um den Namen des Dienstpunkts zu ändern, gehen Sie wie folgt vor:

1. Klicken Sie auf den Namen des Dienstpunktes und drücken Sie die <F2>-Taste.
2. Geben Sie den neuen Namen ein, der bei der Angabe der Interaktionspunkte verwendet wird.

   > ☛ *Weitere Informationen zu den Interaktionspunkten finden Sie unter "Interaktionspunkt eines Elements definieren", Seite 55.*

INTERAKTIONEN VERWALTEN

Eine *Interaktion* stellt den Informationsaustausch zwischen Komponenten einer Architektur dar.

> 🕮 *Eine Interaktion ist ein Vertrag, der in einem bestimmten Kontext zwischen zwei autonomen Entitäten innerhalb oder außerhalb eines Unternehmens abgeschlossen wird. Diese Entitäten können Organisationseinheiten, Aktivitäten oder Prozesse des Unternehmens sein, sowie externe Orgeinheiten eines Unternehmens. Der Inhalt dieses Vertrages wird von einem Austauschvertrag beschrieben.*

Der Inhalt einer Interaktion wird von einem *Austauschvertrag* beschrieben.

> 🕮 *Ein Austauschvertrag ist ein Vertragsmodell zwischen zwei Organisationseinheiten. Dieser Vertrag wird von den Austauschen zwischen der Rolle eines Initiators und einer oder mehreren Rollen der Beitragsleistenden beschrieben.*
>
> ☛ *Weitere Informationen zu den Konzepten eines Austauschvertrags finden Sie unter "Austauschvertrag beschreiben", Seite 58.*

Interaktion erstellen

Um eine Interaktion zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie in der Objektleiste des Diagramms auf die Schaltfläche **Interaktion**.
2. Ziehen Sie eine Verbindungslinie zwischen den beiden kommunizierenden Einheiten.
3. Geben Sie im Fenster zum Hinzufügen von Interaktionen den Austauschvertrag an, der verwendet werden soll.

   > ☛ *Sie können auch einen neuen Austauschvertrag erstellen. Weitere Informationen finden Sie unter "Austauschvertrag ab einer Interaktion erstellen", Seite 59.*
4. Klicken Sie auf **OK**.

Interaktionspunkt eines Elements definieren

Das Prinzip

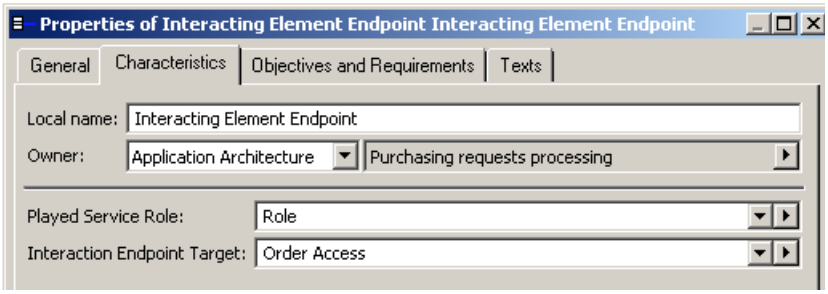
Ein Interaktionspunkt ist ein Element, das eine Interaktion mit einer der kommunizierenden Komponenten verbindet. Mit dem Punkt kann Folgendes angegeben werden:

- der Dienstpunkt oder der Abfragepunkt, der in die Kommunikation eingebunden wird
- Die Rolle, Verbraucher oder Lieferant, die von dem Interaktionspunkt im Austauschvertrag dargestellt werden.

Merkmale des Interaktionspunktes eines Elements definieren

Um die Eigenschaften des Interaktionspunktes eines Elements zu definieren, gehen Sie wie folgt vor:

1. Klicken Sie mit der rechten Maustaste auf die Interaktion neben der kommunizierenden Einheit.
2. Öffnen Sie das Eigenschaftsfenster und klicken Sie auf die Registerkarte **Merkmale**.



3. Wählen Sie **Teilnahme an einer Dienstrolle** aus, d. h. die Rolle des Austauschvertrags, die von dem Interaktionspunkt des Elements übernommen wird.
4. Wählen Sie den **Ziel-Interaktionsendpunkt** aus, d. h. den Dienst- (oder Abfrage-) Punkt, der von der Interaktion betroffen ist.
5. Klicken Sie auf **OK**.

4

Mit Service Design modellieren

In diesem Kapitel wird erklärt, wie Austauschverträge zwischen den Komponenten der IT-Architektur beschrieben werden.

✓ "Austauschvertrag beschreiben", Seite 58
✓ "Austausch beschreiben", Seite 65

AUSTAUSCHVERTRAG BESCHREIBEN

Eine *Interaktion* stellt den Informationsaustausch zwischen den Komponenten einer Architektur dar.

> 🕮 *Eine Interaktion ist ein Vertrag, der in einem bestimmten Kontext zwischen zwei autonomen Entitäten innerhalb oder außerhalb eines Unternehmens abgeschlossen wird. Diese Entitäten können Organisationseinheiten, Aktivitäten oder Prozesse des Unternehmens sein, sowie externe Orgeinheiten eines Unternehmens. Der Inhalt dieses Vertrages wird von einem Austauschvertrag beschrieben.*

Der Inhalt einer Interaktion wird von einem *Austauschvertrag* beschrieben.

> 🕮 *Ein Austauschvertrag ist ein Vertragsmodell zwischen zwei Organisationseinheiten. Dieser Vertrag wird von den Austauschen zwischen der Rolle eines Initiators und einer oder mehreren Rollen der Beitragsleistenden beschrieben.*

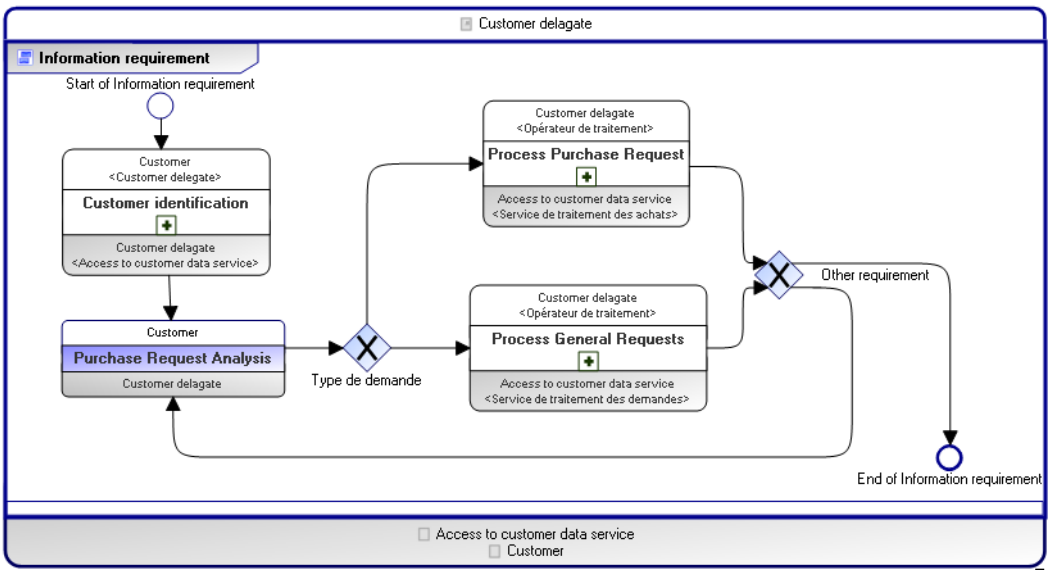
Mit der Option **MEGA Service Design** wird ein Austauschvertrag als Folge einzelner Vorgänge beschrieben, die wie folgt dargestellt werden:

- entweder mit *verwendeten Austauschverträgen*
- oder mit *verwendeten Austauschen*

> 🕮 *Ein verwendeter Austauschvertrag wird mit einem Austauschvertrag verknüpft. Ein Austauschvertrag stellt komplexe Austausche dar.*

> 🕮 *Ein verwendeter Austausch entspricht der Verwendung eines Austausches in einem Austauschvertrag.*

Ein Austauschvertrag wird in einem Diagramm des Austauschvertrags beschrieben. Die Folge der einzelnen Vorgänge wird in einer BPMN-Notation beschrieben.



Beispiel eines Diagramms des Austauschvertrags (BPMN)

✓ "Austauschvertrag ab einer Interaktion erstellen", Seite 59
✓ "Diagramm eines Austauschvertrags erstellen (BPMN)", Seite 60
✓ "Verwendeten Austausch oder verwendeten Austauschvertrag definieren", Seite 60
✓ "Ereignisse erstellen", Seite 61
✓ "Sequenzflüsse bearbeiten", Seite 61
✓ "Die Gateways", Seite 62

Austauschvertrag ab einer Interaktion erstellen

Sie können einen neuen Austauschvertrag erstellen:

- ab einer Bibliothek
- ab einer Interaktion in einem Diagramm

Um einen Austauschvertrag in einem Diagramm ab einer Interaktion zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie in der Objektleiste des Diagramms auf die Schaltfläche **Interaktion** .
2. Ziehen Sie eine Verbindungslinie zwischen den beiden kommunizierenden Einheiten.

3. Klicken Sie im Fenster, in dem die Interaktion hinzugefügt wird, auf den Pfeil rechts neben dem Feld **Austauschvertrag** und wählen Sie **Neu** aus.
   Das Fenster **Erstellung von Austauschvertrag** wird geöffnet.
4. Geben Sie den Namen des Austauschvertrags im Feld **Name** ein.
5. Klicken Sie auf die Schaltfläche **OK**.
   Die Interaktion und der Austauschvertrag wurden erstellt.

Diagramm eines Austauschvertrags erstellen (BPMN)

Mit der Option **MEGA Service Design** wird ein Austauschvertrag mit einem Diagramm eines Austauschvertrags (BPMN) dargestellt.

Um ein Diagramm eines Austauschvertrags (BPMN) ab einer Interaktion zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie mit der rechten Maustaste auf die Interaktion.
2. Wählen Sie den verknüpften Austauschvertrag aus und klicken Sie im Kontextmenü des Vertrags auf **Neu > Diagramm eines Austauschvertrags (BPMN)**.

   ☛ *Wenn Sie in **Windows Front-End** arbeiten, müssen Sie "Diagramm eines Austauschvertrags (BPMN)" in einem spezifischen Fenster auswählen. Stellen Sie sicher, dass das Kontrollkästchen **Diagramm initialisieren** markiert ist und klicken Sie auf **Erstellen**.*

   Das Diagramm wird mit dem Rahmen des Austauschvertrags und zwei *Rollen* geöffnet, die den Verbraucher und den Lieferanten darstellen.

   🕮 *Eine Rolle ist Bestandteil des Objekts, das von ihr beschrieben wird, und kann nicht wiederverwendet werden. Im Anschluss kann sie einer internen oder externen OrgEinheit der Organisation oder einer IT-Komponente zugeordnet werden. Beispiele: Auftraggeber, Reisender.*

Verwendeten Austausch oder verwendeten Austauschvertrag definieren

In einem Diagramm des Austauschvertrags (BPMN) werden die Vorgänge mit folgenden Aktionen beschrieben:

- mit *verwendeten Austauschverträgen*
- mit *verwendeten Austauschen*

  🕮 *Ein verwendeter Austauschvertrag wird mit einem Austauschvertrag verknüpft. Ein Austauschvertrag stellt komplexe Austausche dar.*

  🕮 *Ein verwendeter Austausch entspricht der Verwendung eines Austausches in einem Austauschvertrag.*

Um einen *verwendeten Austauschvertrag* zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie auf die Schaltfläche **Verwendeter n Austauschvertrag** und klicken Sie im Diagramm auf die Form des Austauschvertrags.
   Dann wird das Fenster zum Erstellen einer Aktion angezeigt:
2. Klicken Sie auf den Pfeil rechts neben dem Feld **Spezifikation**.

3. Wählen Sie **Liste** in der Dropdown-Liste aus und wählen Sie den mit dem verwendeten Austauschvertrag verknüpften Austauschvertrag aus.
4. Wählen Sie im Feld **Ab** die Rolle des beschriebenen Austauschvertrags aus, der mit der Rolle Verbraucher des verwendeten Austauschvertrags verbunden ist.
5. Wählen Sie im Feld **Bis** die Rolle des beschriebenen Austauschvertrags aus, der mit der Rolle Lieferant des verwendeten Austauschvertrags verbunden ist.
6. Klicken Sie auf **Beenden**.

Ereignisse erstellen

Die *Ereignisse* "Anfang" und "Ende" sind für die Beschreibung des Dienstes notwendig, der von dem Austauschvertrag übernommen wird.

> 🕮 *Ein Ereignis stellt einen Vorfall oder eine Aktion im System dar, beispielsweise 'Änderung der Kundenadresse'. Es wird von einem Broker verwaltet. Um zu signalisieren, dass sie ein Ereignis produzieren kann, gibt die Anwendung an, dass sie veröffentlicht. Wenn eine Anwendung an einem Ereignis interessiert ist, gibt sie an, dass sie es abonniert.*

Um das Ereignis am Anfang des Austauschvertrags zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie in der Objektleiste auf die Schaltfläche **Ereignis** .
2. Klicken Sie auf den Rahmen des beschriebenen Austauschvertrags. Daraufhin wird das Fenster **Ereignis erstellen** geöffnet.
3. Geben Sie im Feld **Name** "Ende" ein.
4. Wählen Sie im Rahmen "Wahl der Ereignisart" **Ende** aus.
5. Klicken Sie auf **Beenden**.
   Das Ereignis "Ende" des mit "Ende" markierten Austauschvertrags wird im Diagramm angezeigt.

Sequenzflüsse bearbeiten

Ein *Sequenzfluss* ist eine Verbindung in eine bestimmte Richtung, die die chronologische Organisation der verschiedenen Bearbeitungsetappen darstellt.

> 🕮 *Ein Sequenzfluss wird verwendet, um die Reihenfolge ausgeführter Etappen eines Austauschvertrags zu zeigen. Ein Sequenzfluss hat nur eine Quelle und ein Ziel.*

Sequenzfluss erstellen

Um einen *Sequenzfluss* zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie auf die Schaltfläche **Sequenzfluss**.
2. Klicken Sie auf das erste Objekt, das die die Anfangsetappe darstellt und ziehen Sie eine Linie bis zu dem Objekt, das die nachfolgende Etappe darstellt. Halten Sie dabei die Maustaste gedrückt.

Bedingung bei einem Sequenzfluss definieren

Um eine Bedingung bei einem Sequenzfluss zu definieren, gehen Sie wie folgt vor:

3. Klicken Sie mit der rechten Maustaste auf den Sequenzfluss und wählen Sie **Eigenschaften** aus.
4. Klicken Sie in dem nun angezeigten Eigenschaftsfenster auf die Registerkarte **Merkmale**.
5. Klicken Sie auf den Pfeil rechts neben dem Feld **Sequenzflusstyp**.
6. Wählen Sie in der Dropdown-Liste "Bedingt" aus.
7. Geben Sie den Bedingungsausdruck ein und klicken Sie auf **Prädikat**.
8. Klicken Sie auf **OK**.
   Der Text der Bedingung erscheint auf der Verbindung, die folgende Form annimmt: ⟿

Die Gateways

Gemäß der BPMN-Norm werden in der Objektleiste mehrere Typen von *Gateways* aufgeführt.

> 🕮 *Ein Gateway ist ein Modellierungselement, das zur Kontrolle der Sequenzflüsse verwendet wird, die in einem Prozess zusammen- oder auseinanderlaufen.*

Um die wichtigsten Anwendungsfälle besser verstehen zu können, unterscheiden wir:

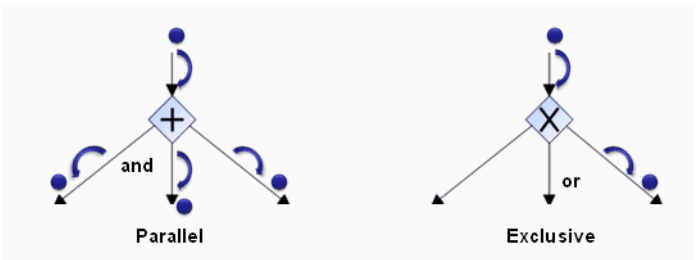
- Gateways am Ausgang einer Etappe
- Gateways am Eingang einer Etappe

Gateways am Ausgang einer Etappe

Bei einem **Exklusiven** Gateway kann nur eine Ausgangsverzweigung unter den verschiedenen möglichen Verzweigungen gewählt werden. Das Gateway kann gewählt werden:

- abhängig von den **Daten**, die für die Bearbeitung verfügbar sind
- abhängig von den **Ereignissen**, die während der Bearbeitung auftreten können

Bei einem **Parallelen** Gateway werden alle Ausgangsverzweigungen gleichzeitig bearbeitet.



Bei einem **Inklusiven** Gateway können ein oder mehrere Ausgangsverzweigungen unter den verschiedenen möglichen Verzweigungen gewählt werden.

Ein **Komplexes** Gateway entspricht einer Kombination der oben genannten Gateways.

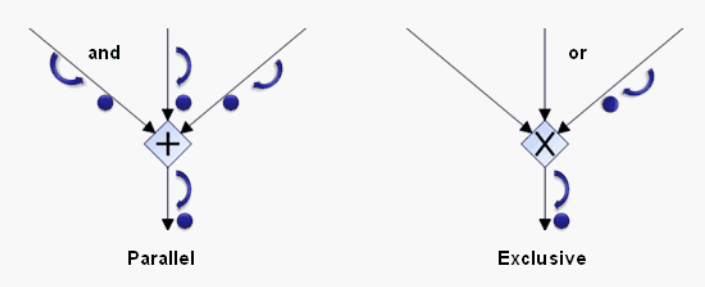
☛ *Wenn das Gateway erstellt ist, kann der Typ des Gateways im Eigenschaftsfenster geändert werden.*

Die Gateways am Eingang einer Etappe

Am Eingang einer Etappe stellt ein Gateway den Punkt dar, an dem die Sequenzflüsse am Eingang zusammenlaufen.

Bei einem **Exklusiven** Gateway wird die Etappe des Prozesses ausgeführt, sobald eine der Verzweigungen aktiv ist.

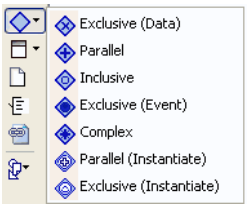
Bei einem **Parallelen** Gateway werden alle Eingangsverzweigungen gleichzeitig bearbeitet.



Gateway erstellen

Um ein Gateway zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie in der Objektleiste auf die Schaltfläche **Gateways** und wählen Sie den betreffenden Gateway-Typ aus.



2. Klicken Sie in das Diagramm.
   Das Gateway wird im Diagramm mit der ausgewählten Form angezeigt.

Gateway ändern

Um ein Gateway zu ändern, gehen Sie wie folgt vor:

1. Klicken Sie mit der rechten Maustaste auf das Gateway und wählen Sie **Eigenschaften** aus.
   Das Eigenschaftsfenster wird angezeigt.

2. Ändern Sie in der Registerkarte **Merkmale** den Namen oder den Typ des Gateways. Folgende Gateway-Typen werden angezeigt:
   - **Komplex**: bei der Ausführung kann eine komplexe Wegekombination verwendet werden
   - **Exklusiv (Daten-basiert)**: Bei der Ausführung können eine oder mehrere Wege verwendet werden, je nachdem, welcher Datenwert zur Verfügung steht. Es handelt sich hier um das Standard-Gateway
   - **Exklusiv (Anfang)**: Die Ausführung wird vom ersten Ereignis ausgelöst, das eintritt. Die anderen werden ignoriert.
   - **Exklusiv (Ereignis-basiert)**: Die Ausführung kann einen oder mehrere Wege verwenden, je nach den Ereignissen, die eintreten.
   - **Inklusiv**: Die Ausführung kann gleichzeitig mehrere Wege verwenden.
   - **Parallel**: Die Ausführung verwendet gleichzeitig mehrere parallele Wege.
   - **Parallel (Anfang)**: Die Ausführung wird vom ersten Ereignis ausgelöst, das eintritt. Die anderen Ereignisse während der Ausführung werden ebenfalls berücksichtigt.
3. Klicken Sie auf **OK**.

AUSTAUSCH BESCHREIBEN

Der Inhalt einer Interaktion wird von einem *Austauschvertrag* beschrieben.

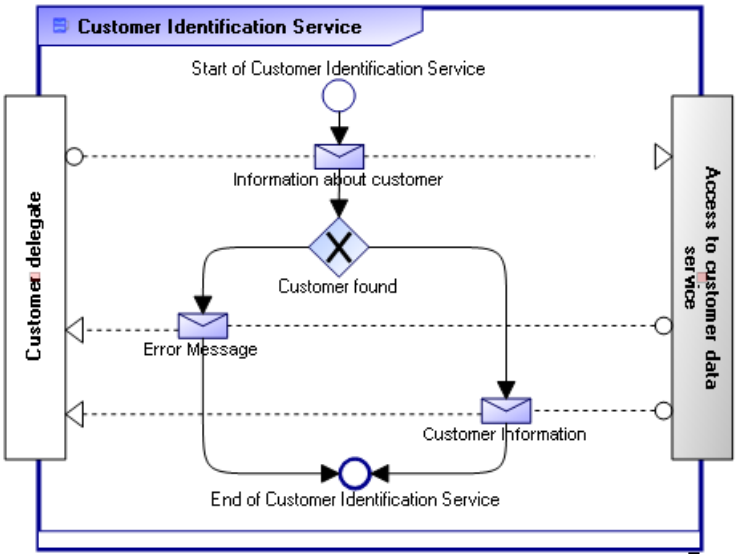
> *Ein Austauschvertrag ist ein Vertragsmodell zwischen zwei Organisationseinheiten. Dieser Vertrag wird von den Austauschen zwischen der Rolle eines Initiators und einer oder mehreren Rollen der Beitragsleistenden beschrieben.*

Mit der Option **MEGA Service Design** kann ein Austauschvertrag mit einer Folge von verwendeten Austauschen oder Austauschverträgen beschrieben werden. Ein *verwendeter Austausch* wird mit einem *Austausch* verknüpft.

> *Ein verwendeter Austausch entspricht der Verwendung eines Austausches in einem Austauschvertrag.*

> *Ein Austausch beschreibt den Austausch von Nachrichtenflüssen zwischen den Teilnehmern.*

Ein *Austausch* wird von einem Diagramm eines Austauschs beschrieben, das die Folgen ausgetauschter Nachrichten darstellt.




```
Der Protokollablauf des Dienstes der Kundenidentifikation
beginnt mit dem Versand von Informationen, mit denen der
Kunde identifiziert wird. Wenn der Kunde nicht gefunden
wird, wird eine Fehlermeldung ausgegeben. Andernfalls
werden die Auskünfte über den Kunden gesendet
(Kundenidentifikation, Status der Bestellungen usw.).
```

✓ "Austausch erstellen", Seite 66
✓ "Austausch beschreiben", Seite 66

Austausch erstellen

Sie können einen *Austausch* ab einer Bibliothek oder einem Diagramm eines Austauschvertrags (BPMN) erstellen.

> 🕮 *Ein Austausch beschreibt den Austausch von Nachrichtenflüssen zwischen den Teilnehmern.*

Um einen *Austausch* ab einem Diagramm eines Austauschvertrags (BPMN) zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie auf die Schaltfläche **Verwendeter Austausch** und klicken Sie im Diagramm auf die Form des beschriebenen Austauschvertrags.
   Daraufhin wird das Fenster zur Erstellung eines Austauschs angezeigt.
2. Klicken Sie auf den Pfeil rechts neben dem Feld **Spezifikation** und wählen Sie in der Dropdown-Liste **Erstellen** aus.
   Daraufhin wird das Fenster zur Erstellung eines Austauschs angezeigt.
3. Geben Sie den **Namen** des Austauschs ein.
4. Klicken Sie auf **OK**, um dieses Fenster zu schließen.
   Der Austausch wird automatisch erstellt.
5. Fahren Sie mit der Erstellung des verwendeten Austauschs fort.

Austausch beschreiben

Austauschdiagramm (BPMN) erstellen

Ein *Austausch* wird von einem Diagramm eines Austauschs beschrieben, das die Folgen ausgetauschter Nachrichten darstellt.

Um ein Austauschdiagramm zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie mit der rechten Maustaste auf einen **Austausch** und klicken Sie auf **Neu** > **Austauschdiagramm (BPMN)**.

   > ☛ *Wenn Sie in **Windows Front-End** arbeiten, müssen Sie "Austauschdiagramm (BPMN)" in einem spezifischen Fenster auswählen. Stellen Sie sicher, dass das Kontrollkästchen **Diagramm initialisieren** markiert ist und klicken Sie auf **Erstellen**.*

   Daraufhin wird das Diagramm geöffnet. Der Rahmen des Austauschs ist positioniert. Die beiden Rollen (Verbraucher und Lieferant) sind erstellt.

Nachrichtenfluss mit Inhalt erstellen

Sie können die *Nachrichtenflüsse* und ihren *Inhalt* angeben, die zwischen den beiden Rollen des Austauschs ausgetauscht werden.

> 🕮 *Ein Nachrichtenfluss stellt den Informationsfluss in einem Austauschvertrag dar. Ein Fluss transportiert den Inhalt.*

> 🕮 *Mit Inhalt wird der Inhalt einer Nachricht oder eines Ereignisses unabhängig von seiner Struktur bezeichnet. Die Struktur wird durch ein mit dem Inhalt verbundenen XML-Schema dargestellt. Ein Inhalt kann von mehreren Nachrichten verwendet werden, da er nicht mit einem*

Sender oder einem Empfänger verknüpft ist. Pro Nachricht oder pro Ereignis gibt es jeweils immer nur einen Inhalt, aber derselbe Inhalt kann von mehreren Nachrichten oder Ereignissen verwendet werden.

Um einen Nachrichtenfluss und seinen Inhalt zu erstellen, gehen Sie wie folgt vor:

1. Klicken Sie im Austauschdiagramm auf die Schaltfläche **Nachrichtenfluss mit Inhalt.**
2. Klicken Sie auf die Rolle, die den Sender des Nachrichtenflusses darstellt, und ziehen Sie eine Linie bis zu der Rolle, die den Empfänger des Nachrichtenflusses darstellt. Halten Sie dabei die Maustaste gedrückt. Das Fenster **Erstellung von Nachrichtenfluss mit Inhalt** wird geöffnet.
3. Wählen Sie in der Dropdown-Liste des Felds **Inhalt** den Inhalt aus, der mit dem Nachrichtenfluss verbunden werden soll.
   Der Nachrichtenfluss wird mit seinem Inhalt im Diagramm angezeigt.

STANDARDBERICHTE VON MEGA SYSTEM ORIENTED IT ARCHITECTURE VERWENDEN

MEGA System Oriented IT Architecture bietet Funktionen zur Analyse einer Architektur. Diese Funktionen verwenden Berichte. Mit den Berichtsvorlagen kann der Austausch zwischen den Agenten analysiert werden, die in der Architektur eingesetzt werden. Mit der **MEGA**-Suite können Objekt-Sets aus dem Repository in den Berichten gruppiert werden und die Interaktion dieser Objekte geprüft werden.

☛ *Weitere Informationen zum Funktionsprinzip der Berichte finden Sie im Handbuch **MEGA Common Features** im Kapitel "Berichte generieren".*

Die Liste der standardmäßig von **MEGA System Oriented IT Architecture** vorgeschlagenen Berichtsvorlagen wird angezeigt:

- ✓ "Übereinstimmung der Austausche zwischen Agenten". Weitere Informationen finden Sie unter "Ausgleich der Austausche zwischen Agenten", Seite 72.
- ✓ "Übereinstimmung der Austausche zwischen den Anwendungen". Dieser Bericht ist mit dem Bericht "Übereinstimmung der Austausche zwischen Agenten" identisch. Das Thema des Berichts kann nur vom Typ Anwendung sein.
- ✓ "Ausgleich der Austausche zwischen Agenten". Weitere Informationen finden Sie unter "Übereinstimmung der Austausche zwischen Agenten", Seite 70.
- ✓ "Ausgleich der Austausche zwischen Anwendungen". Dieser Bericht ist mit dem Bericht "Ausgleich der Austausche zwischen Agenten" identisch. Das Thema des Berichts kann nur vom Typ Anwendung oder logische Anwendung sein.
- ✓ "Kreuztabelle der Interaktionen von Diensten", Seite 74.

ÜBEREINSTIMMUNG DER AUSTAUSCHE ZWISCHEN AGENTEN

Der Bericht über die Übereinstimmung der Austausche zwischen Agenten enthält eine Synthese der Informationen, die von den Agenten erhalten wurden, die für den Bericht parametriert wurden.

Die Agenten sind in den verschiedenen Benutzungskontexten mit der Außenwelt über Interaktionen verbunden, deren verbundene Austauschverträge den Austausch der Nachrichten beschreiben.

Dieser Bericht erstellt für jeden Agenten eine Liste der erhaltenen Interaktionen, deren Austauschvertrag von Nachrichten beschrieben wird.

```
Der nachfolgende Bericht führt die Nachrichten auf, die
über die erhaltenen Interaktionen von den Anwendungen
"Einkauf eCommerce" und "Anrufmanagement" und den
Anwendungsarchitekturen "Zahlungsmanagement" und "Plattform
der Einkaufsverwaltung" ausgetauscht werden.
```

2. Agent Exchange Compliance

[Add a comment for this chapter]

This table shows for each interaction the compliance between the expected exchange and the information element effectiv

0% of the exchanges are compliant.
12 exchanges are considered.
12 exchanges are missing (100%).

Legend:
OK (Expected information element has effectively been exchanged)
Missing (Expected information element has not been exchanged)
? Not expected (Information element has effectively been exchanged but it was not expected)

Exchange Compliance				
Expected Exchanges	Sources	Targets	Information Elements	E E
Information requirement (Information requirement)	Call Center (Call Center)	Call Management (Call Management)	Information about customer	
			Customer Information	
			Error Message	
Customer identification (Customer identification)	Call Management (Call Management)	Internet Purchasing (Internet Purchasing)	Information about customer	
			Customer Information	
			Error Message	
Customer identification	Customer	Internet Purchasing	Information about	

Berichtparameter

Dieser Bericht übernimmt nur den Parameter **Agent**. Dieser Parameter ist obligatorisch. Die Metaklassen sind akzeptiert:

- OrgEinheit
- Anwendung
- Anwendungssystem
- Architektur logischer Anwendungen
- Logische Anwendung
- Geschäftsbereich

AUSGLEICH DER AUSTAUSCHE ZWISCHEN AGENTEN

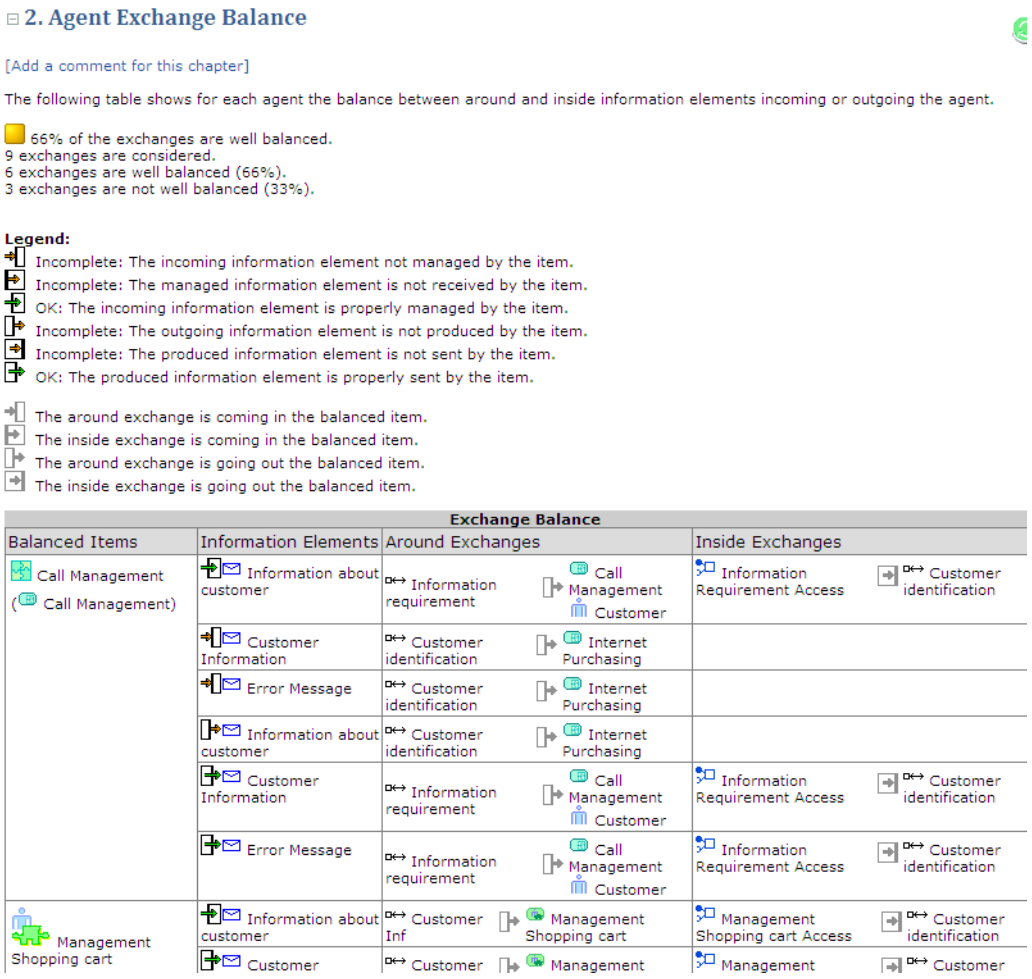
Dieser Bericht enthält eine Synthese der Informationen, die von den parametrierten Agenten für den Bericht erhalten und bearbeitet werden.

Die Agenten sind in den verschiedenen Benutzungskontexten mit der Außenwelt über Interaktionen verbunden, deren verbundene Austauschverträge den Austausch der Nachrichten beschreiben.

Ein Agent wird als "ausgeglichen" betrachtet, wenn die Beschreibung der internen Funktionsweise des Agenten alle mit der Außenwelt ausgetauschten Nachrichten übernimmt.

Der nachfolgende Bericht führt einen ausgeglichenen Agenten, "Warenkorb des Kunden verwalten" und einen nicht

```
ausgeglichenen Agenten "Warenkorb des Kunden verwalten"
auf, dessen interne Struktur nicht beschrieben wird.
```



2. Agent Exchange Balance

[Add a comment for this chapter]

The following table shows for each agent the balance between around and inside information elements incoming or outgoing the agent.

66% of the exchanges are well balanced.
9 exchanges are considered.
6 exchanges are well balanced (66%).
3 exchanges are not well balanced (33%).

Legend:

Incomplete: The incoming information element not managed by the item.
Incomplete: The managed information element is not received by the item.
OK: The incoming information element is properly managed by the item.
Incomplete: The outgoing information element is not produced by the item.
Incomplete: The produced information element is not sent by the item.
OK: The produced information element is properly sent by the item.

The around exchange is coming in the balanced item.
The inside exchange is coming in the balanced item.
The around exchange is going out the balanced item.
The inside exchange is going out the balanced item.

Exchange Balance

Balanced Items	Information Elements	Around Exchanges	Inside Exchanges
Call Management (Call Management)	Information about customer	Information requirement — Call Management / Customer	Information Requirement Access — Customer identification
	Customer Information	Customer identification — Internet Purchasing	
	Error Message	Customer identification — Internet Purchasing	
	Information about customer	Customer identification — Internet Purchasing	
	Customer Information	Information requirement — Call Management / Customer	Information Requirement Access — Customer identification
	Error Message	Information requirement — Call Management / Customer	Information Requirement Access — Customer identification
Management Shopping cart	Information about customer	Customer Inf — Management Shopping cart	Management Shopping cart Access — Customer identification
	Customer	Customer — Management	Management — Customer

Berichtparameter

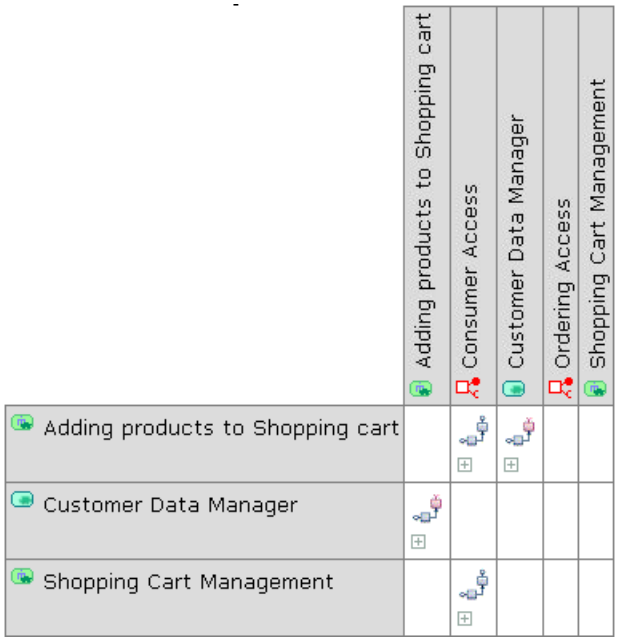
Dieser Bericht übernimmt nur den Parameter **Agent**. Dieser Parameter ist obligatorisch. Die Metaklassen sind akzeptiert:

- OrgEinheit
- Anwendung
- Anwendungssystem
- Architektur logischer Anwendungen
- Logische Anwendung
- Geschäftsbereich

KREUZTABELLE DER INTERAKTIONEN VON DIENSTEN

Dieser Bericht enthält die Liste der Interaktionen, die in den Agenten verwendet werden, welche im Thema des Berichts parametriert werden.

Der Bericht



Die in den Zeilen aufgeführten Objekte sind die "Verbraucher" der Interaktionen und Urheber der Dienstanträge.

☛ *In den Zeilen werden die Objekte aufgeführt, die direkt mit den Agenten verbunden sind, welche das Thema des Berichts sind, sowie die Objekte, die mit diesen Objekten verbunden sind.*

☛ *Nur die Objekte (und Unterobjekte) mit einem definierten Interaktionspunkt, der die Rolle des Verbrauchers übernimmt, werden in dem Bericht aufgeführt.*

Die Objekte in den Spalten sind die "Lieferanten" der Interaktionen und bieten die Antworten auf die Dienstanträge.

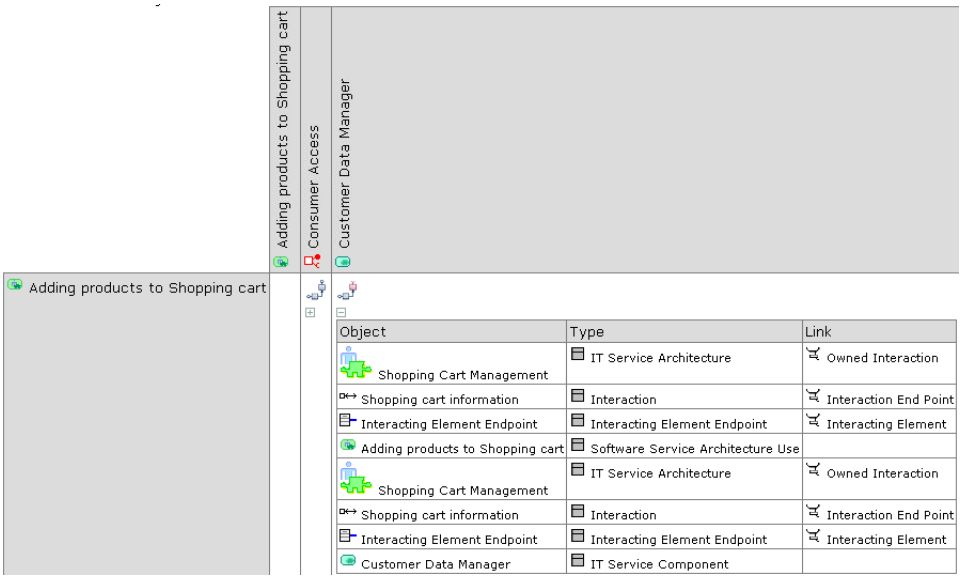
☛ *In den Spalten werden die Objekte aufgeführt, die direkt mit den Agenten verbunden sind, welche das Thema des Berichts sind, sowie die Objekte, die mit diesen Objekten verbunden sind.*

☛ *Nur die Objekte (und Unterobjekte) mit einem definierten Interaktionspunkt, der die Rolle des Lieferanten übernimmt, werden in dem Bericht aufgeführt.*

☛ *Weitere Informationen zu den Interaktionspunkten finden Sie unter "Interaktionspunkt eines Elements definieren", Seite 55.*

Details des Austauschs

Wenn Sie im Kästchen der Kreuztabelle auf die Schaltfläche klicken, erhalten Sie Detailangaben der Interaktion, die die beiden Objekte verbindet.



Object	Type	Link
Shopping Cart Management	IT Service Architecture	Owned Interaction
Shopping cart information	Interaction	Interaction End Point
Interacting Element Endpoint	Interacting Element Endpoint	Interacting Element
Adding products to Shopping cart	Software Service Architecture Use	
Shopping Cart Management	IT Service Architecture	Owned Interaction
Shopping cart information	Interaction	Interaction End Point
Interacting Element Endpoint	Interacting Element Endpoint	Interacting Element
Customer Data Manager	IT Service Component	

Die Informationen werden in folgender Reihenfolge aufgeführt:

- Interaktion, die mit dem Objekt "Verbraucher" verbunden ist und hintereinander mit folgenden Elementen beschrieben wird:
  - Objekt, das den Kontext darstellt: Objekt, das das Objekt "Verbraucher " und die Interaktion enthält
  - Interaktion, in der das Objekt "Verbraucher" seinen Antrag sendet
  - Interaktionspunkt, über den das Objekt "Verbraucher" seinen Antrag sendet
  - Objekt "Verbraucher" (ebenfalls in der Zeile).
- Interaktion, die mit dem Objekt "Lieferant" verbunden ist
  - Objekt, das den Kontext darstellt: Objekt, das das Objekt "Lieferant" und die Interaktion enthält
  - Interaktion, in der das Objekt "Lieferant" seinen Antrag empfängt
  - Interaktionspunkt, über den das Objekt "Lieferant" seinen Antrag empfängt
  - Objekt "Lieferant" (ebenfalls in der Spalte).

Berichtparameter

Hier werden die Eingabedaten des Berichts definiert.

Einstellung	Parametertyp	Einschränkungen
Agent	OrgEinheit Anwendung Anwendungssystem Architektur logischer Anwendungen Logische Anwendung Geschäftsbereich	Mindestens ein obligatorisches Thema. Die Agenten werden kumuliert.
Teil eines filternden Agenten	OrgEinheit Anwendung Anwendungssystem Architektur logischer Anwendungen Logische Anwendung Geschäftsbereich	Nicht obligatorisch
Filternder Agent	OrgEinheit Anwendung Anwendungssystem Architektur logischer Anwendungen Logische Anwendung Geschäftsbereich	Nicht obligatorisch

GLOSSAR

Das Glossar stellt die wichtigsten Konzepte vor, die in **MEGA Service Oriented Architecture Design** verwendet werden.

Abfragepunkt	Ein Abfragepunkt ist ein Austauschpunkt, über den ein Agent bei potenziellen Lieferanten einen Dienst beantragt.
Agenttyp	Ein Agenttyp ist die Klasse einer Einheit, die aktiv an Prozessen teilnimmt und Services nutzt oder liefert. Geschäftsbereiche, organisatorische Einheiten, Anwendungen und Server sind Agenttypen.
Anwendung	Eine Anwendung besteht aus Software-Komponenten, die im Rahmen der IT-Entwicklungen eine konsistente Einheit bilden.
Anwendungskomponente	Eine Anwendungskomponente beschreibt die Rolle einer Anwendung in einer zusammengesetzten Beziehung mit einem Anwendungssystem oder einer übergeordneten Anwendung.
Anwendungsprozess	Ein Systemprozess ist die ausführbare Darstellung eines Prozesses. Folgende Elemente formalisieren einen Systemprozess: Workflow-Elemente, zu erfüllende Tasks während der Bearbeitung, algorithmische Elemente eines Sequenzflusses von Tasks, mit Teilnehmern ausgetauschte Informationsflüsse.
Anwendungssystem	Ein Anwendungssystem ist eine Montage aus anderen Anwendungssystemen, Anwendungen und Endbenutzern, die mit den Anwendungskomponenten zusammenwirken, um eine oder mehrere Funktionalitäten durchzuführen.
Architektur logischer Anwendungen	Die Architektur logischer Anwendungen ist eine Montage aus anderen Architekturen, logischen Anwendungen und Endbenutzern, die mit den Anwendungskomponenten zusammenwirken, um eine oder mehrere Funktionalitäten durchzuführen.

Austausch

Ein Austausch beschreibt den Austausch von Nachrichtenflüssen zwischen den Teilnehmern.

Austauschvertrag

Ein Austauschvertrag ist ein Vertragsmodell zwischen zwei Organisationseinheiten. Dieser Vertrag wird von den Austauschen zwischen der Rolle eines Initiators und einer oder mehreren Rollen der Beitragsleistenden beschrieben.

Bibliothek

Bibliotheken sind Gruppierungen von Objekten, mit denen der Inhalt eines MEGA-Repositorys in mehrere unabhängige Teile unterteilt wird. Mit Bibliotheken lassen sich virtuelle Repository-Partitionen erstellen. Vor allem können zwei Objekte aus verschiedenen Bibliotheken denselben Namen tragen.

Data Store

Ein Data Store bietet einen Mechanismus an, mit dem Daten aktualisiert oder abgerufen werden, die auch über die Reichweite des aktuellen Prozesses hinaus bestehen bleiben. In einem Data Store werden die Daten der Flüsse am Eingang gespeichert und über einen oder mehrere Flüsse am Ausgang weitergeleitet.

Diagramm

Ein Diagramm wird verwendet, um die mit dem Grafiktool erstellten Grafiken zu speichern, z.B. Flowcharts, Organigramme usw.

Dienstpunkt

Ein Dienstpunkt ist ein Austauschpunkt, über den ein Agent potenziellen Kunden einen Dienst anbietet.

Endbenutzer

Ein Endbenutzer stellt eine OrgEinheit dar, die an den Grenzen eines Anwendungssystems oder einer Architektur logischer Anwendungen mitwirkt.

Ereignis

Ein Ereignis stellt einen Vorfall oder eine Aktion im System dar, beispielsweise 'Änderung der Kundenadresse'. Es wird von einem Broker verwaltet. Um zu signalisieren, dass sie ein Ereignis produzieren kann, gibt die Anwendung an, dass sie veröffentlicht. Wenn eine Anwendung an einem Ereignis interessiert ist, gibt sie an, dass sie es abonniert.

Externes Data Store

Ein externes Data Store ist ein Data Store, das innerhalb und außerhalb des beschriebenen Systems verwendet wird.

Fluss

Ein Nachrichtenfluss stellt den Informationsfluss in einem Austauschvertrag dar. Ein Fluss transportiert den Inhalt.

Funktionalität

Eine Funktionalität ist ein Dienst, der von einer Organisationseinheit erwartet wird, damit sie ihre Arbeit ausführen kann. Diese Funktionalität kann allgemein im Rahmen einer Aktivität oder genauer zur Ausführung eines bestimmten Arbeitsschritts notwendig sein. Die Funktionalität kann von einer Anwendung bereitgestellt werden, wenn es sich um eine rechnergestützte Funktionalität handelt.

Gateway

Ein Gateway ist ein Modellierungselement, das zur Kontrolle der Sequenzflüsse verwendet wird, die in einem Prozess zusammen- oder auseinanderlaufen.

Inhalt

Mit Inhalt wird der Inhalt einer Nachricht oder eines Ereignisses unabhängig von seiner Struktur bezeichnet. Die Struktur wird durch ein mit dem Inhalt verbundenen XML-Schema dargestellt. Ein Inhalt kann von mehreren Nachrichten verwendet werden, da er nicht mit einem Sender oder einem Empfänger verknüpft ist. Pro Nachricht oder pro Ereignis gibt es jeweils immer nur einen Inhalt, aber derselbe Inhalt kann von mehreren Nachrichten oder Ereignissen verwendet werden.

Interaktion

Eine Interaktion ist ein Vertrag, der in einem bestimmten Kontext zwischen zwei autonomen Entitäten innerhalb oder außerhalb eines Unternehmens abgeschlossen wird. Diese Entitäten können Organisationseinheiten, Aktivitäten oder Prozesse des Unternehmens sein, sowie externe Orgeinheiten eines Unternehmens. Der Inhalt dieses Vertrages wird von einem Austauschvertrag beschrieben.

IT-Service

Ein IT-Service ist das Teilelement einer Anwendung, das dem Endbenutzer dieser Anwendung im Rahmen seiner Arbeit zur Verfügung gestellt wird.

IT-Service-Komponente

Eine IT-Service-Komponente stellt die Tatsache dar, dass der verwendete IT-Service eine Rolle in der beschriebenen Anwendung spielt.

Komponente einer logischen Anwendung

Die Komponente einer logischen Anwendung beschreibt die Rolle einer logischen Anwendung in einer zusammengesetzten Beziehung mit einer übergeordneten logischen Anwendung oder einer übergeordneten Architektur logischer Anwendungen.

Komponente einer Plattformanwendung

Komponente einer Plattformanwendung stellt die Tatsache dar, dass die verwendete Technologie in der beschriebenen Anwendung die Rolle einer Plattform übernimmt. Beispielsweise ist Tomcat eine Plattformkomponente für SageCRM.

Komponente eines Agenttyps

Eine Komponente eines Agenttyps beschreibt die Rolle in einer zusammengesetzten Beziehung, mit der sie mit einem anderen Agenttyp verbunden wird.

Komponente eines Agenttyps ist eine abstrakte Klasse.

Logische Anwendung

Eine logische Anwendung ist das Teilelement einer Anwendung, das dem Endbenutzer dieser Anwendung im Rahmen seiner Arbeit zur Verfügung gestellt wird. Die logischen Anwendungen stellen die Aufteilung der Anwendungen in Form funktionaler elementarer Einheiten dar. Eine logische Anwendung ist eine kohärente und nicht teilbare Bearbeitungseinheit, die eine Gruppe von Nachrichten und Ereignissen koordiniert, um im Informationssystem eine Task durchzuführen. Im Anwendungssystem entspricht die logische Anwendung einer funktionellen Einheit, die weitaus elementarer ist.

Logisches Data Store	Ein logisches Data Store stellt die Verwendung der Daten über Anwendungssysteme dar, ohne zu berücksichtigen, wie der Zugriff konkret implementiert wird.
Lokales Data Store	Ein lokales Data Store ist ein Data Store, das ausschließlich innerhalb des beschriebenen Systems verwendet wird.
Nachricht	Eine Nachricht stellt einen Fluss innerhalb des Unternehmens oder zwischen dem Unternehmen und seiner Umgebung dar. Im Allgemeinen handelt es sich um einen Informationsfluss, wie z.B. eine Bestellung oder eine Rechnung. Der Einfachheit halber wird ein Finanzfluss, wie z.B. die Zahlung des Kunden, oder ein Materialfluss, beispielsweise die Lieferung eines Produkts, ebenfalls als Nachricht dargestellt.
Organisationseinheit	Eine OrgEinheit stellt eine Person oder eine Personengruppe dar, die in Prozesse oder in das Informationssystem eines Unternehmens eingebunden ist. Eine OrgEinheit kann unternehmensintern oder -extern sein: Eine interne Organisationseinheit stellt ein Element der Organisation eines Unternehmens dar, wie beispielsweise eine Direktion, eine Abteilung oder einen Arbeitsplatz. Sie wird auf einer mehr oder weniger genauen Ebene definiert, je nachdem, wie präzise die Angaben über die Organisation sein müssen (siehe Organisationseinheitstyp). Beispiel: Finanzdirektion, Vertrieb, Marketingabteilung, Vertriebsmitarbeiter. Eine externe Organisationseinheit stellt eine Einrichtung dar, die Flüsse mit dem Unternehmen austauscht. Beispiel: Kunde, Lieferant, Administration.
Personal	Das Personal entspricht konfigurierten organisatorischen Ressourcen, um einige Fähigkeiten im Rahmen einer Ressourcenarchitektur zu unterstützen.
Physisches Data Store	Ein physisches Data Store stellt die Implementierung eines logischen Data Stores dar.
Physisches RDB-Data Store	Ein Relationale Datenbank-Data Store ist ein Data Store in Form von Tabellen oder Tabellenansichten.
Profil	Eine Rolle ist ein Teilnehmer einer Interaktion, eines Workflows oder eines Prozesses. Sie kann einem Initiator entsprechen, d. h. dem Auftraggeber eines Dienstes, oder einen Zulieferer darstellen, der eine Dienst-externe Bearbeitung ausführt. Eine Rolle ist Bestandteil des Objekts, das von ihr beschrieben wird, und kann nicht wiederverwendet werden. Im Anschluss kann sie einer internen oder externen OrgEinheit der Organisation oder einer IT-Komponente zugeordnet werden. Beispiele: Auftraggeber, Reisender.
RE-Data Store	Ein Entität/Relation-Data Store beschreibt die Daten einer Datenbank in Form von Entitäten oder Klassen.

Realisierung	Eine Realisierung beschreibt die Beziehung zwischen einer logischen und einer physischen Entität, mit der sie umgesetzt wird. Die physische Entität ergibt eine Liste der logischen Entitäten, die von ihr implementiert werden.
Sequenzfluss	Ein Sequenzfluss wird verwendet, um die Reihenfolge ausgeführter Etappen eines Austauschvertrags zu zeigen. Ein Sequenzfluss hat nur eine Quelle und ein Ziel.
Server	Ein Server ist eine Hardware-Ressource, die über eine Datenbank verfügen kann, auf der eine Anwendung ausgeführt werden kann.
Technologie	Eine Technologie ist eine Definition oder ein Format, das von einer Normungsorganisation zugelassen wurde oder von der Industrie als Standard akzeptiert wird.
Verwendeter Austausch	Ein verwendeter Austausch entspricht der Verwendung eines Austausches in einem Austauschvertrag.
Verwendeter Austauschvertrag	Ein verwendeter Austauschvertrag wird mit einem Austauschvertrag verknüpft. Ein Austauschvertrag stellt komplexe Austausche dar.
Verwendung einer Architektur logischer Anwendungen	Die Nutzung einer Architektur logischer Anwendungen beschreibt die Rolle einer Architektur logischer Anwendungen in einer zusammengesetzten Beziehung mit einer übergeordneten Architektur logischer Anwendungen.
Verwendung eines Anwendungssystems	Die Nutzung eines Anwendungssystems beschreibt die Rolle eines Anwendungssystems in einer zusammengesetzten Beziehung mit einem übergeordneten Anwendungssystem.